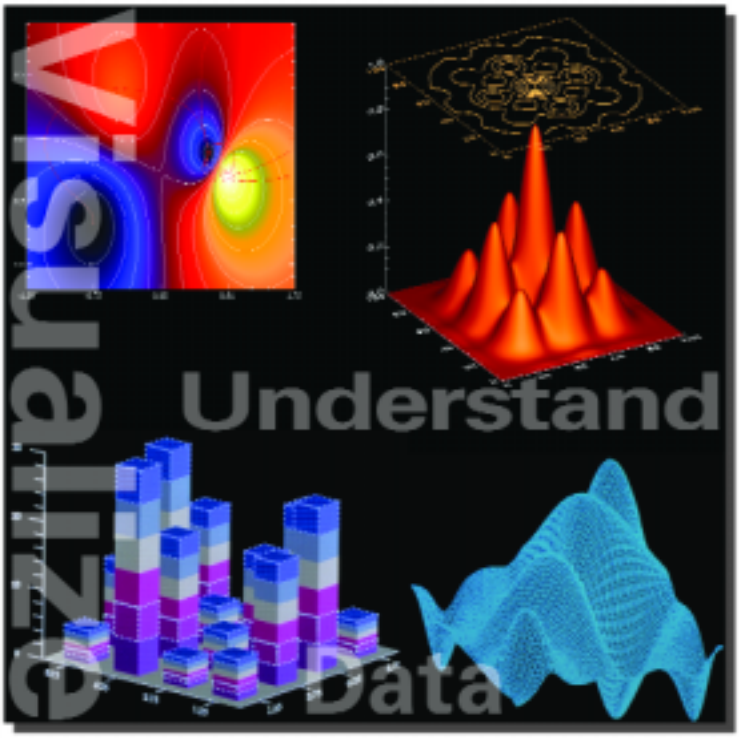
TS-WAVE™ 3.0



User's Guide

HELPING CUSTOMERS **SOLVE** COMPLEX PROBLEMS

Visual Numerics, Inc.

Visual Numerics, Inc.
2500 Wilcrest Drive
Suite 200
Houston, Texas 77042-2579
United States of America
713-784-3131
800-222-4675
(FAX) 713-781-9260
http://www.vni.com
e-mail: info@boulder.vni.com

Visual Numerics, Inc. (France) S.A.R.L.
Tour Europe
33 place des Corolles
Cedex 07
92049 PARIS LA DEFENSE
FRANCE
+33-1-46-93-94-20
(FAX) +33-1-46-93-94-39
e-mail: info@vni-paris.fr

Visual Numerics International, Ltd.
Suite 1
Centennial Court
East Hampstead Road
Bracknell, Berkshire
RG 12 1 YQ
UNITED KINGDOM
+01-344-458-700
(FAX) +01-344-458-748
e-mail: info@vniuk.co.uk

Visual Numerics, Inc.
7/F, #510, Sect. 5
Chung Hsiao E. Rd.
Taipei, Taiwan 110 ROC
+886-2-727-2255
(FAX) +886-2-727-6798
e-mail: info@vni.com.tw

Visual Numerics International GmbH
Zettachring 10
D-70567 Stuttgart
GERMANY
+49-711-13287-0
(FAX) +49-711-13287-99
e-mail: info@visual-numerics.de

Visual Numerics Japan, Inc.
Gobancho Hikari Building, 4th Floor
14 Gobancho
Chiyoda-Ku, Tokyo, 102
JAPAN
+81-3-5211-7760
(FAX) +81-3-5211-7769
e-mail: vda-sprt@vnij.co.jp

VIsual Numerics S.A. de C.V.
Cerrada de Berna 3, Tercer Piso
Col. Juarez
Mexico, D.F. C.P. 06600
Mexico

Visual Numerics, Inc., Korea
Rm. 801, Hanshin Bldg.
136-1, Mapo-dong, Mapo-gu
Seoul 121-050
Korea

Table of Contents

Preface

This user's guide explains how to use TS-WAVE, a powerful tool for analyzing and plotting data quickly, from Visual Numerics. This guide contains the following parts:

- **Chapter 1, *Getting Started*** — This chapter is a tutorial and a user's guide that covers the most commonly used functions of TS-WAVE.
- **Chapter 2, *Developer's User Reference*** — This chapter describes briefly the primary functions necessary to access the DH data structures.
- **Chapter 3, *GUI Reference*** — A complete guide to the menus and dialog boxes in TS-WAVE.
- **TS-WAVE Index**

Intended Audience

TS-WAVE is designed as an end-user application for test engineers who need to subset, analyze, and plot data quickly, efficiently, and accurately. Because TS-WAVE functionality is provided through a graphical user interface, it is relatively easy to learn. No programming experience is required to use TS-WAVE. Some knowledge of PV-WAVE and PV-WAVE VDA Tools is helpful, but not required.

Typographical Conventions

- The following notation means "select the **Graph Object** function from the **Create** menu":

  Select **Create=>Graph Object**.

- In this user's guide, "click" means to press and release the left mouse button quickly; "drag" means to hold down the left mouse button while moving the mouse.
- Keyboard keys are written like this <Keyname>. For example, <Return>, <Control>, and <X>.
- <Control>-<X> means to press the <X> key while holding down the <Control> key.

Technical Support

If you have problems installing, unlocking, or running your software, contact Visual Numerics Technical Support by calling:

Office Location	Phone Number
Corporate Headquarters Houston, Texas	713-954-6439
Boulder, Colorado	303-939-8920
France	+33-1-46-93-94-20
Germany	+49-711-13287-0
Japan	+81-3-5211-7760
Korea	+82-2-3273-2633
Mexico	+52-5-514-9730
Taiwan	+886-2-727-2255
United Kingdom	+01-344-458-700

Users outside the U.S., France, Germany, Japan, Korea, Mexico, Taiwan, and the U.K. can contact their local agents.

Please be prepared to provide the following information when you call for consultation during Visual Numerics business hours:

- Your license number, a six-digit number that can be found on the packing slip accompanying this order. (If you are evaluating the software, just mention that you are from an evaluation site.)
- The name and version number of the product. For example, TS-WAVE 3.0.
- The type of system on which the software is being run. For example, SPARCstation, IBM RS/6000, HP 9000 Series 700.
- The operating system and version number. For example, HP-UX 10.2 or IRIX 6.2.
- A detailed description of the problem.

FAX and E-mail Inquiries

Contact Visual Numerics Technical Support staff by sending a FAX to:

Office Location	FAX Number
Corporate Headquarters	713-781-9260
Boulder, Colorado	303-245-5301
France	+33-1-46-93-94-39
Germany	+49-711-13287-99
Japan	+81-3-5211-7769
Korea	+82-2-3273-2634
Mexico	+52-5-514-4873
Taiwan	+886-2-727-6798
United Kingdom	+01-344-458-748

or by sending E-mail to:

Office Location	E-mail Address
Boulder, Colorado	`support@boulder.vni.com`
Houston, Texas	`support@houston.vni.com`

France	support@vni-paris.fr
Germany	support@visual-numerics.de
Japan	vda-sprt@vnij.co.jp
Korea	support@vni.com.kr
Taiwan	support@vni.com.tw
United Kingdom	support@vniuk.co.uk

Electronic Services

General e-mail	info@boulder.vni.com
Support e-mail	support@boulder.vni.com
World Wide Web	www.vni.com www.vni.com/tswave
Anonymous FTP	ftp.boulder.vni.com
FTP Using URL	ftp://ftp.boulder.vni.com/VNI/
PV-WAVE Mailing List:	Majordomo@boulder.vni.com
To subscribe include:	subscribe pv-wave YourEmailAddress
To post messages	pv-wave@boulder.vni.com

1

CHAPTER

Getting Started

After completing this tutorial, you will be able to:

- start and stop TS-WAVE
- load a data file
- subset a data file
- create and manipulate graphs
- create a header
- plot your data
- modify the plot's appearance
- create a contour
- open another data source
- zoom in on data
- process your data using standard and user functions
- create a tabular data file
- save your work
- create an XY plot

The tutorial takes about an hour to complete.

At the end of this chapter, some additional topics are discussed:

- creating a pick file
- saving and using a template
- creating a batch file

- adding user-defined functions

Introduction to TS-WAVE

TS-WAVE is an application to display and analyze data. With the use of TS-WAVE's graphical user interface, users can subset data, create and customize plots on a page, choose which parameters to plot, run analysis functions on data, generate tabular data files, and perform other analysis tasks.

TS-WAVE New Features

- TS-WAVE's improved DataHandlers now have the ability to manage and display data from multiple files, which is referred to as multi-source functionality. This enhancement greatly simplifies the level of implementation required to integrate proprietary file formats into TS-WAVE.

  Now, you can compare previously tested data with current test data; view multiple data sets at the same time; compare data from one quarter to another quarter simultaneously in a side-by-side view; compare predictive or simulation data to actual collected data; and much more. These enhancements make TS-WAVE's functionality much more accurate and precise.

- A new graphic contour object was added, a user-requested enhancement that provides interactive graphical functionality to our users. TS-WAVE's new graphic contour object extends plotting beyond the traditional X-Y plotting to allow 2-D contour plotting.

Collecting Data for TS-WAVE

The manner in which your data is collected varies from site to site. TS-WAVE is capable of dealing with almost any data file format. Your data may be collected in a different manner, and it may or may not have time-tag information associated with the parameters. TS-WAVE is distributed with a sample data file from a flight test group. This data file will be used as you progress through the remainder of this Introduction.

During a typical flight test, aircraft data is recorded in digital format on an airborne data acquisition system recorder. The system records data anytime the recorder is operating. During some flights the recorder is operated for the entire flight. To aid in data management, there is a "run number" parameter incorporated into the data. This run number provides a method for separating the flight data into smaller segments.

After the flight test, the test data is removed from the recorder and an engineering data file is created. This data file can be read into TS-WAVE, subsetted (for example, by run number), analyzed, and plotted. Security measures can be implemented so that data files are available, on a read-only basis, to multiple TS-WAVE users over a network.

TS-WAVE – Usage

Users of TS-WAVE can plot data parameters on the *y*-axis against time on the *x*-axis or an independent (x-axis) variable, the independent variable can be any valid data parameter. Users can plot an unlimited number of x*y*-axis variables simultaneously on a single graph. It also lets users create tabular data files (a convenient way to subset your data) and batch files.

Using TS-WAVE

In this tutorial, you will use most of the major features of TS-WAVE.

Starting and Stopping TS-WAVE

Starting TS-WAVE on a Windows System

If TS-WAVE is installed properly, an icon appears on your desktop: **TSWAVE**. Double **click** on the TS-WAVE icon to start the application.

The main window of TS-WAVE opens, as shown in *Figure 1-1*. In addition, a PV-WAVE command window opens.

Starting TS-WAVE on a UNIX System

If TS-WAVE is installed properly, the shell script is created in the PV-WAVE installation directory. Check with your local system administrator if you are unsure of the PV-WAVE installation directory at your site.

Step 1 Source the setup file for PV-WAVE, where *maindir* is the path to the main installation directory (e.g., `/usr/local/vni`):

From a C-Shell:

source *maindir*/**wave/bin/wvsetup**

From a Bourne- or Korn-Shell:

. *maindir*`/wave/bin/wvsetup.sh`

Step 2 Once you have sourced wvsetup (or run wvsetup.sh), enter:

tswave

The main window of TS-WAVE opens, as shown in *Figure 1-1*.

TS-WAVE also allows a number of execution-time options. To examine the list of available options, enter:

tswave -h

Stopping TS-WAVE

Step 1 From the **File** menu, select **Exit**.

NOTE From now on, we will use the notation **Menu=>Function** to describe a menu selection. For example: Select **File=>Exit**.

Step 2 (*UNIX only*) Type `EXIT` at the `WAVE>` prompt in the PV-WAVE command window to exit the PV-WAVE session.

CAUTION Before you **exit** TS-WAVE be sure to **save** your work. TS-WAVE **does not** automatically save your work; and it **does not** warn you to save your work.

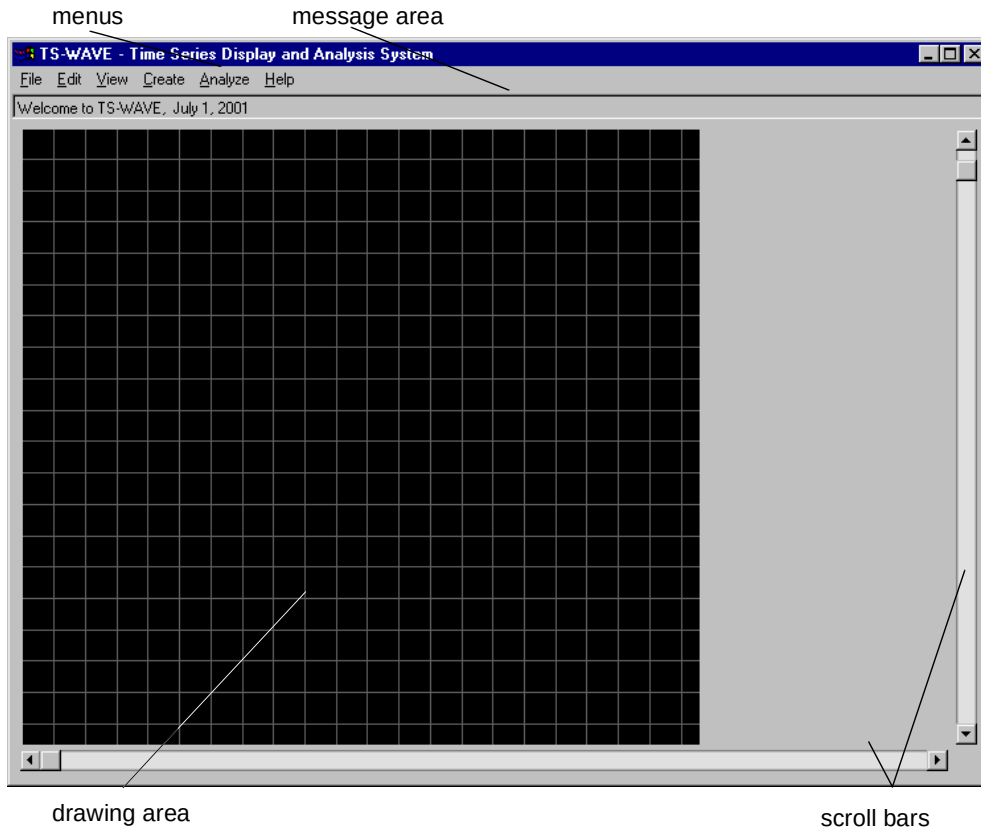


Figure 1-1 The TS-WAVE Main Window

Opening and Subsetting a Test Data File

Start by opening the sample data file. This file consists of dozens of parameters that reflect measurements taken during five different runs.

Step 1 If TS-WAVE is not already started, start it now.

Step 2 Select **File=>Open**. The Open Data File dialog box appears as shown in *Figure 1-2*.

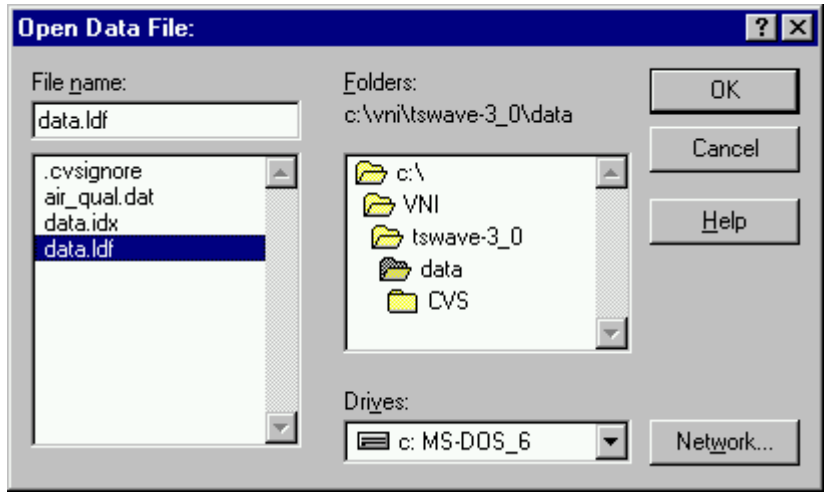


Figure 1-2 Open Date File dialog box

Step 3 Select the *data.ldf* file and click **OK**:

(Windows) `<install_dir>\data\data.ldf`

(UNIX) `<install_dir>/data/data.ldf`

where `<install_dir>` is the main installation directory.

The actual directory path might differ from the one shown here, depending on where you installed TS-WAVE.

The Name and Source ID dialog box appears, as shown in *Figure 1-3*.

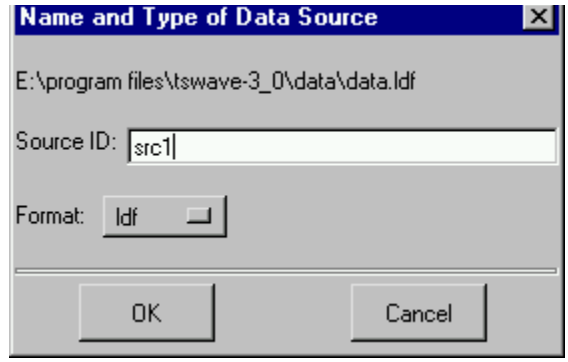


Figure 1-3 The Name and Type of Data Source dialog box is used to assign a source id and select a file format.

Step 4 Notice the default Source ID type is *src*1, and the file format type is *ldf*, click **OK**.

The Select Data Run dialog box appears, as shown in *Figure 1-4*.

NOTE This file consists of 4.5 Mb of data. The file contains an ASCII header that defines the time slices, a list of run numbers, a list of parameters, and 8000 binary records taken in five runs at specific time increments.

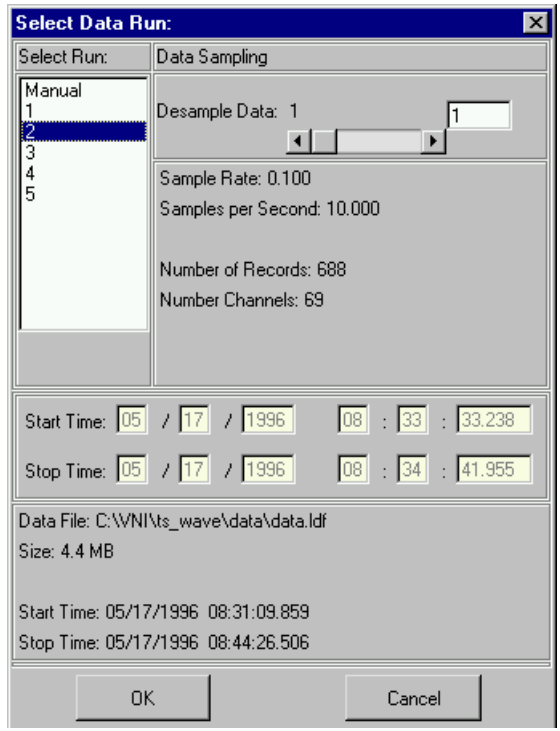


Figure 1-4 The Select Data Run dialog box is used to select the run, or runs, to load from the data file.

Step 5 **Click** on run number **3** to select it.

TIP Only the data from this run number will be read into TS-WAVE. To read the entire file of **test data** at once, **click** on **Manual**. With Manual selected, you can edit the **Run Start** and **Run Stop** text fields to read specific time intervals.

The following information is displayed in the dialog:

Sample Rate:	0.100
Samples per Seconds:	10.000
Number of Records:	674
Number of Channels:	69

Step 6 Select a **Desampling** data rate of **60** by either moving the slider or entering `60` and pressing <Return> in the text field above and to the right of the slider. The desampling rate specifies the interval at which records are retrieved. A rate of 1 retrieves all records. There are 674 records in this run, so with a desampling rate of 60, we are only grabbing around 10 data points.

NOTE Start Time and Stop Time are grayed out but displays the range for Run 3. The text below displays information about the datafile: the name, the size, and the top Start/Stop times for the entire field.

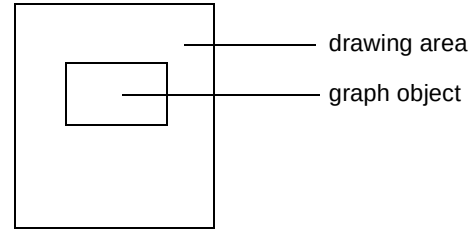
Step 7 Click **OK** to process and retrieve the information that you entered.

Creating a Graph

TS-WAVE data is plotted on 2D graphs, where the *x*-axis by default is the Time axis, and the *y*-axis is the axis on which data parameters are plotted.

Step 1 Select **Create=>Graph Object**.

Step 2 Position the mouse pointer in the drawing area where you want the upper left-hand corner of the graph to be located. We recommend placing the pointer at about coordinate (5, 22). Then, hold down the left mouse button while dragging the mouse pointer to about coordinate (18, 13) to define the position of the lower-right corner of the graph. When you release the mouse button, the graph is displayed in the drawing area. The following schematic shows approximately where to position the graph in the drawing area:



Selecting, Resizing, and Removing a Graph

You can resize or delete a graph object, but first you must select it. It is a good idea to learn how to select graph objects (and other kinds of graphical objects) now, because you will select objects often when using TS-WAVE.

Step 1 Select **Edit=>Object Select**. This command allows you to select a graph object.

TIP The message area just below the Graph menu says “Object Selection”. This message area typically displays information on the current task that you are performing.

Step 2 **Click** anywhere in the graph.

NOTE “Handles” appear at points on the perimeter of the graph. These handles indicate that the graph is selected. When selected, the graph can be resized, cut, deleted, or copied.

Step 3 Resize the graph. Move the pointer over one of the handles, hold down the left mouse button, and drag the handle to resize the graph. Release the mouse button to complete the resizing.

TIP When you press down the mouse button, the pointer changes to a double arrow. If the pointer does not change, be sure that the pointer is just inside the handle of the graph when you press down the mouse button.

Step 4 Select **Edit=>Cut**. The graph disappears from the drawing area. Actually, the **Cut** function places the graph in temporary storage called the clipboard. If you had selected **Edit=>Delete**, the graph would be permanently removed.

TIP Notice on the Edit menu that most of the functions have keyboard shortcuts. For the **Cut** function, the keyboard shortcut is **Ctrl-X**. This means, that you can cut the currently selected object either by selecting the function from the menu or by using the shortcut (in this case, hold down the <Control> key and press the <X> key).

Step 5 Select **Edit=>Paste** (or use the shortcut: **Ctrl-V**) to paste the graph back in the drawing area.

TIP To select multiple objects: hold down the <Shift> key and click on the objects you wish to select; or, choose **Edit=>Select All** to select all objects in the drawing area at once; or, press and hold the left mouse button and drag the pointer around the objects you wish to select.

TIP To deselect an object, hold down the <Shift> key and click on the object. To deselect all objects, choose **Edit=>Deselect All**.

Moving a Graph

To move a graph, make sure it is your currently selected object. Then, position the pointer inside the graph, and hold down the left mouse button (this causes the pointer to change its shape). You can then drag the graph to its new position. Try it if you wish.

Creating Another Graph

Create a second graph just below the first graph. (As before, use the function **Create=>Graph Object**.) Position the upper left corner at about coordinate (5, 11) and the lower right corner at about (18, 3).

The drawing area with two graphs is shown in *Figure 1-5*.

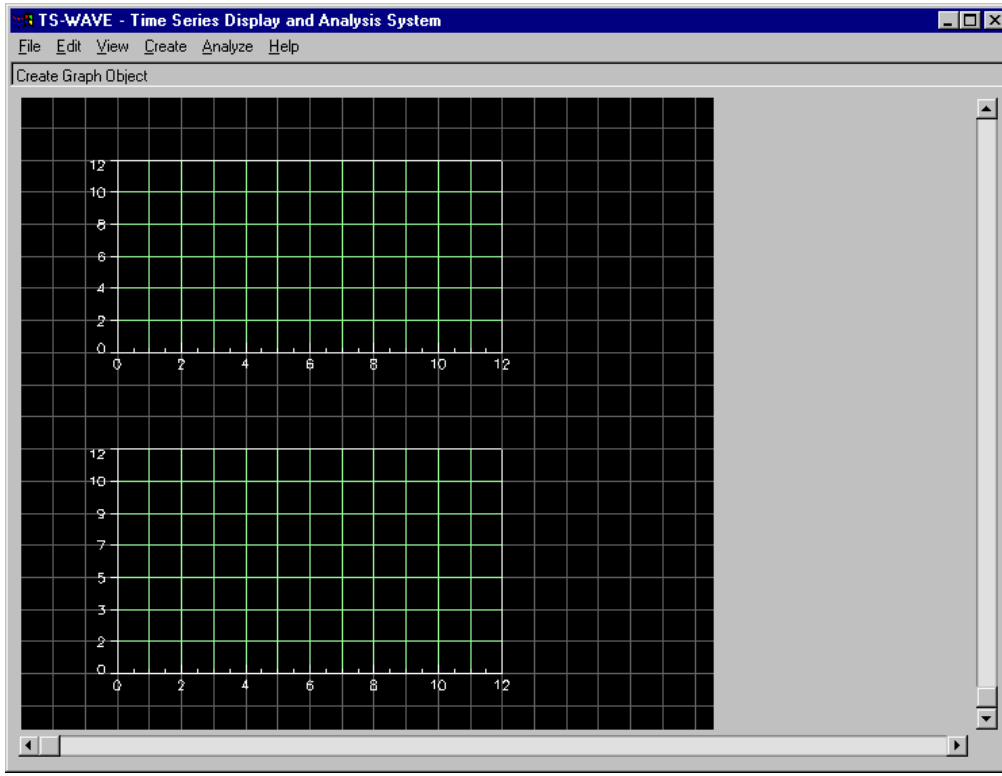


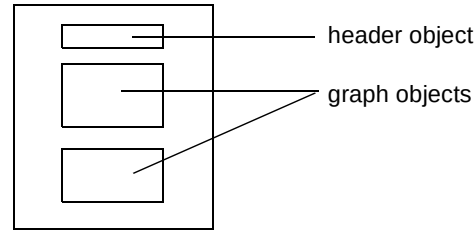
Figure 1-5 Two graph objects were added to the drawing area.

Adding a Header

A header is a text box that you can add to the drawing area. Headers are created just like graphs.

Step 1 Select **Create=>Header Object**.

Step 2 In the same manner as you created the graph objects, use the mouse to create a header box just above the top graph. Position the upper left corner at about coordinate (5, 27), and the lower right coordinate at about (18, 24). The following schematic shows the approximate position of the graphs and header in the drawing area:



Step 3 Select **Edit=>Object Select**.

Step 4 Double **click** anywhere inside the header box to bring up the Header Attributes Interface dialog box, shown in *Figure 1-6*. This dialog is used to specify the contents of the header and to modify the appearance of the text (font, size, color, and so on).

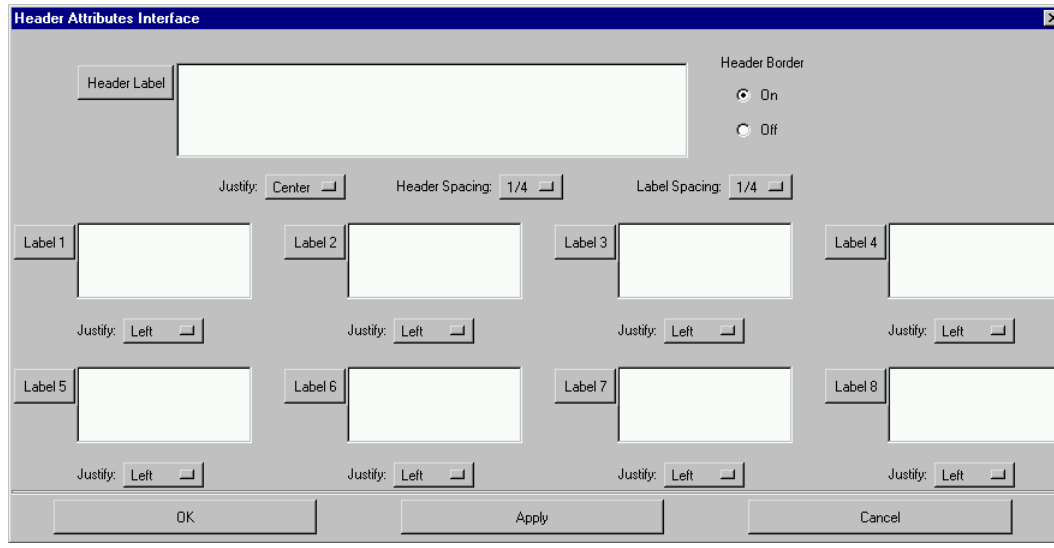


Figure 1-6 The Header Attributes Interface dialog box.

Step 5 In the **Header Label** field, enter TS-WAVE `Tutorial`.

Step 6 Select **Center** from the **Justify** menu just below the Header Label text field.

Step 7 Click the **Header Label** button to bring up the Advanced Label Editing Interface dialog box. This dialog lets you change the font, text size, position, and other text characteristics. Change the **Print Size** to `2.0`, and

click **OK** to dismiss the Advanced Label Editing Interface dialog. For more information on this dialog box, see *Advanced Label Editing Interface Dialog Box* on page 102.

Step 8 In the **Label 1** text field, enter the following text:

```
ALTB Avg: %%AVG(src1:ALTB):(F8.2)%%
```

This line demonstrates how you can embed PV-WAVE functions in a TS-WAVE header label. When a line with this syntax is included in a header, the specified calculation is made whenever a new dataset is loaded into the TS-WAVE session. In this case, the average of the Boom Altitude parameter is calculated and displayed in the header. For more information embedding PV-WAVE functions in header labels, see *Header Attributes Interface Dialog Box* on page 87.

Step 9 Click **OK** in the Header Attributes Interface dialog box.

Figure 1-7 shows the drawing area after the addition of the header.

TIP If you want to modify the header text, simply double click again inside the header box. If you wish, experiment now with adding labels and with modifying the appearance of the text.

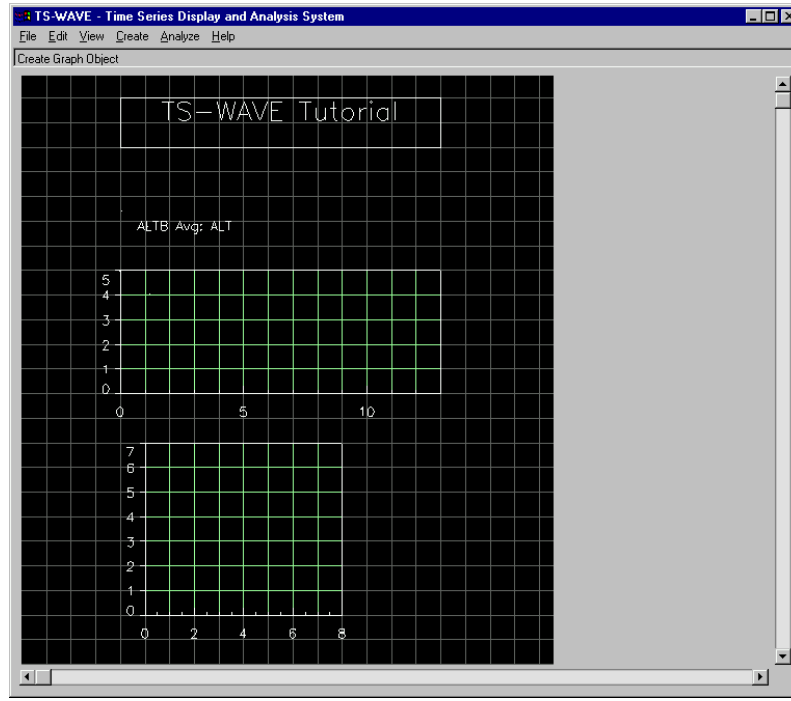


Figure 1-7 The drawing area contains two graphs and a header.

Graphing Data Parameters

You already opened a data file, created two graph objects, and a header. Now it is time to plot your data.

Step 1 Select **Edit=>Object Select**.

Step 2 Double click inside the top graph object. The Graph Attributes Interface dialog box appears, as shown in *Figure 1-8*.

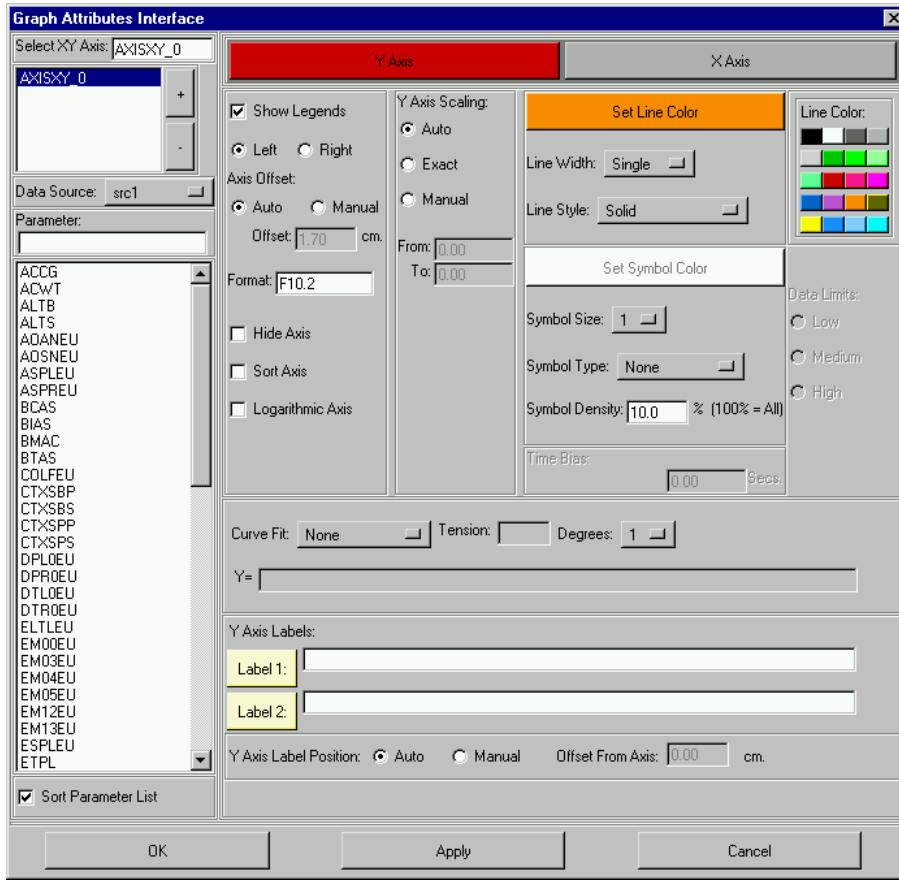


Figure 1-8 The Graph Attributes Interface dialog box.

TIP This dialog box has a lot of functions; however, its primary purpose is for selecting the parameters that you wish to plot.

Step 3 In the list of parameters, locate the COLFEU parameter and click on it. Notice the parameter name is now highlighted.

Step 4 In the **Label 2** text field, enter `Elevator Force`. This is the long name for the COLFEU parameter.

Step 5 Near the top left of this interface, locate the small square button that has a '+' (plus symbol) in the center of the button. Click on this button to add a second **XY Axis** button.

Step 6 Find the parameter ESPLEU and click on it.

TIP To find a specific parameter in the parameter list, you can type the parameter name (if you know it) in the **Parmeter** text field of the dialog box. The parameter is automatically located and highlighted for you.

Step 7 In the **Label 2** text field, enter `Elevator Position`. This is the long name for the ESPLEU parameter.

Step 8 Click **OK**. The dialog closes, and both parameters are plotted in the graph.

NOTE With the addition of the second parameter to the graph, a second *y*-axis is added to the left of the first *y*-axis.

TIP If necessary, move or resize the graph so that you can see both *y*-axes.

Step 9 Following the same procedure as for the first graph, add the **ALTB** parameter to the lower graph. This is the Boom Altitude parameter.

Now you have plotted three variables from the original data file. Next, let's modify the appearance of one of the plots.

Modifying the Plot's Appearance

Step 1 If the Object Selection mode is not active, select **Edit=>Object Select**. (If it is active, "Object Selection" appears in the left corner of the message bar.)

Step 2 Double click on the top graph. The Graph Attributes Interface dialog box appears.

Step 3 Select **AxisXY_0** in the "Select XY Axis" list. This makes the first plotted parameter, **COLFEU**, active.

Notice that the upper portion of the dialog contains several functions for modifying the graph. For now, we'll just modify a couple of settings. For detailed information

on all of the settings in this dialog box, see *Graph Attributes Interface Dialog Box* on page 92.

Step 4 Using the option menus provided (toward the upper right of the dialog box), change the **Line Color** to **Red**, the **Line Style** to **Long Dashes**, and the **Line Width** to **Single**.

Step 5 Click **OK**, and notice how the graph has changed to reflect the new settings.

Creating a Contour

Lets you add a contour object to the drawing area. The general procedure for adding a contour object includes these steps:

Step 1 Select **Create=>Contour Object**.

Step 2 Position the mouse pointer in the drawing area where you want the upper left-hand corner of the contour object to be located. When you release the mouse button, the contour object is displayed in the drawing area.

Step 3 Select **Edit=>Object Select**.

Step 4 Double click anywhere inside the contour object box to bring up the Contour Attributes Interface dialog box as in *Figure 1-9*. Use this dialog to specify the appearance of the contour object such as drawing method, plot attributes, level of custom settings and more. Let's plot a contour using the 'Follow' method. By default 'Follow' is selected for you.

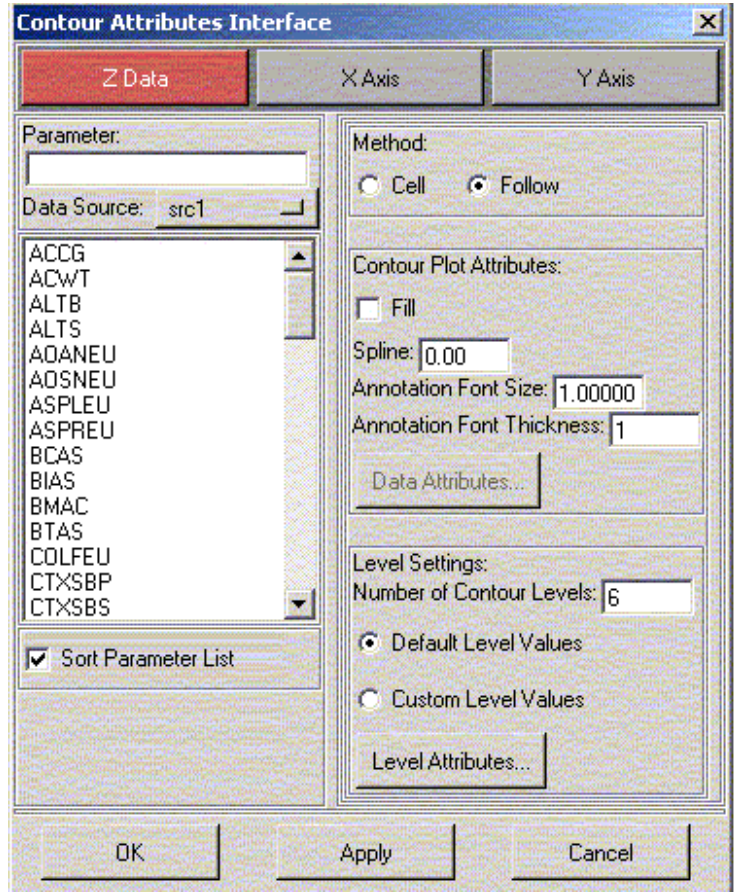


Figure 1-9 The Contour Attributes Interface dialog box (for more information, see *Contour Attributes Interface Dialog Box* on page 98).

Step 5 Let's assign a parameter to the X, Y and Z axes. Select the **X Axis** tab and then choose **COLFEU** from the parameter list; select the **Y Axis** tab and choose **ESPLEU** from the parameter list; finally select the **Z Data** tab and choose **ASPLEU** from the parameter list. Click **APPLY**. Parameters are now assigned to each axis. Watch as the plot appears in the drawing area.

Step 6 Click **OK** to dismiss the Attributes Interface dialog box.

Step 7 Let's try a few settings on the **Z Data** tab. Change the "Number of Contour Levels" to 10. Next, change the "Annotation Font Size" to 3.0000.

Click on Level Attributes to change the color of a contour: the Level Attributes dialog box appears (see *Level Attributes Dialog* on page 102). Select a contour level from the list. Then click Line/Fill Color, and select a color. Click **DISMISS** to close the Level Attributes dialog box. Click **Apply** and watch the contour object as it appears in the drawing area. Click **OK** to dismiss the Contour Attributes Interface dialog box.

For a more detailed discussion on contouring, see *Contour Attributes Interface Dialog Box* on page 98.

Opening Another Data Source

You can change the run number at any time. By changing the run number, the new run data is read from the data file and plotted immediately. Furthermore, any functions specified in the header are recalculated and redisplayed.

Step 1 Select **File=>Open**. The Open Data File dialog box appears.

Step 2 Choose the *data.ldf* file and click **OK**.

Step 3 Notice ***src*2** appears in the Source ID text field. Click **OK**. The Select Data Run dialog box appears as shown in *Figure 1-4*.

Step 4 Select run 1, 2, 4, or 5 from the run list. If you wish, you can change the desampling Rate or other parameters in this dialog box.

Step 5 Click **OK** to plot parameters from the new run.

Step 6 Double click on the top graph. The Graph Attributes Interface dialog box appears. Near the top left, right below the Select XY Axis field, locate the square button with the '+' symbol and click on it. Now locate the Data Source drop-down menu and choose **src2**, then select **COLFEU** from the Parameter list and click **APPLY** to update the graph. Finally, click **OK** to dismiss the dialog box.

NOTE The new parameters are immediately plotted, and the average boom altitude is recalculated and displayed in the header.

Zooming in on Data

The TS-WAVE Data Zoom feature lets you see plotted data up close. To enlarge the image of your data, do the following:

Step 1 **Select** one or more graph objects displaying at least one or more parameters.

Step 2 **Open** the Data Zoom dialog box by selecting **View=>Data Zoom**. The Data Zoom dialog box appears, as shown in *Figure 1-10*.

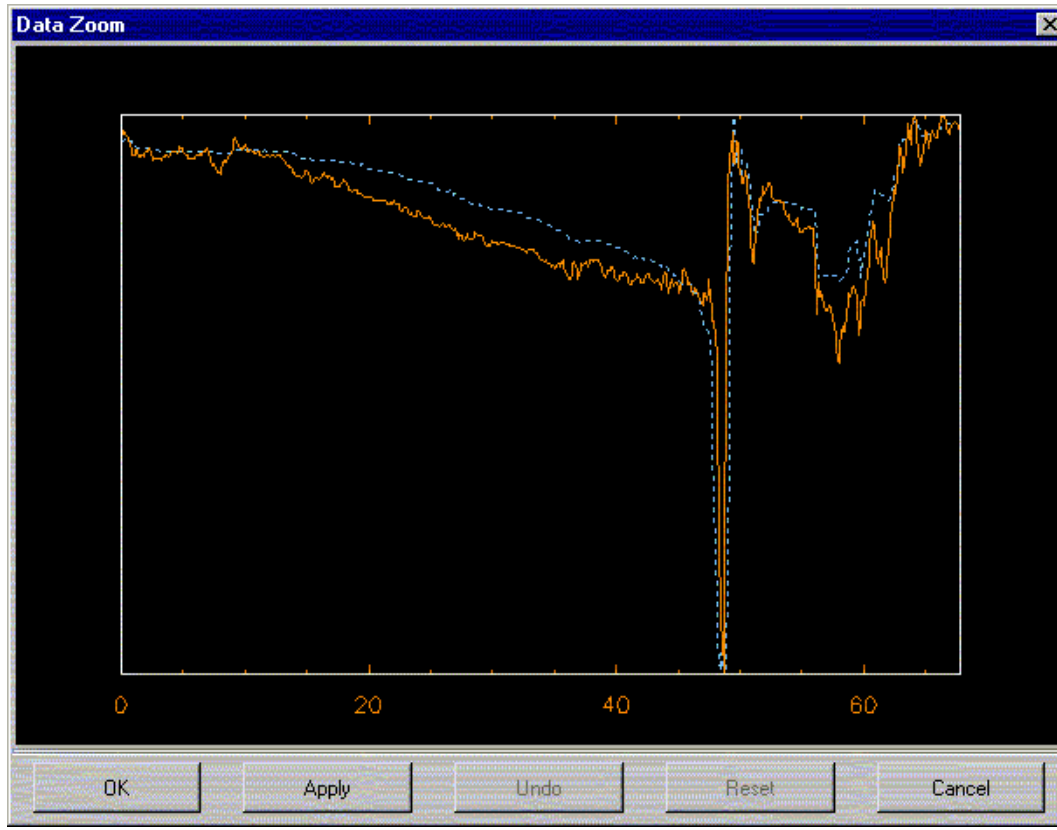


Figure 1-10 The Data Zoom dialog box.

Step 3 **Click** and **drag** your left mouse button in the Data Zoom window to zoom in on a "specific" range on the plot. Notice the rubber band box being drawn. Release the mouse, and you will be zoomed in on the "specific" range defined by the rubber band box.

Step 4 Once the data has been zoomed use the **Up/Down/Left/Right** arrow keys to pan up and down or left to right on the plot. For more details on keyboard operations functionality see *Data Zoom* on page 77.

Step 5 Click **OK** to update the zoomed range of the plotted data on the selected graph object, or **click Cancel** to exit without saving the current zooms/ scales. For more details on this functionality see *Data Zoom* on page 77.

Figure 1-11 displays plotted data that has been zoomed in on.

NOTE The **Undo** dialog button reverts to the previous zoomed range(s) or original range if it was the first zoom. For more details on functionality of the **OK/Apply/ Undo/Reset** dialog buttons see *Data Zoom* on page 77.

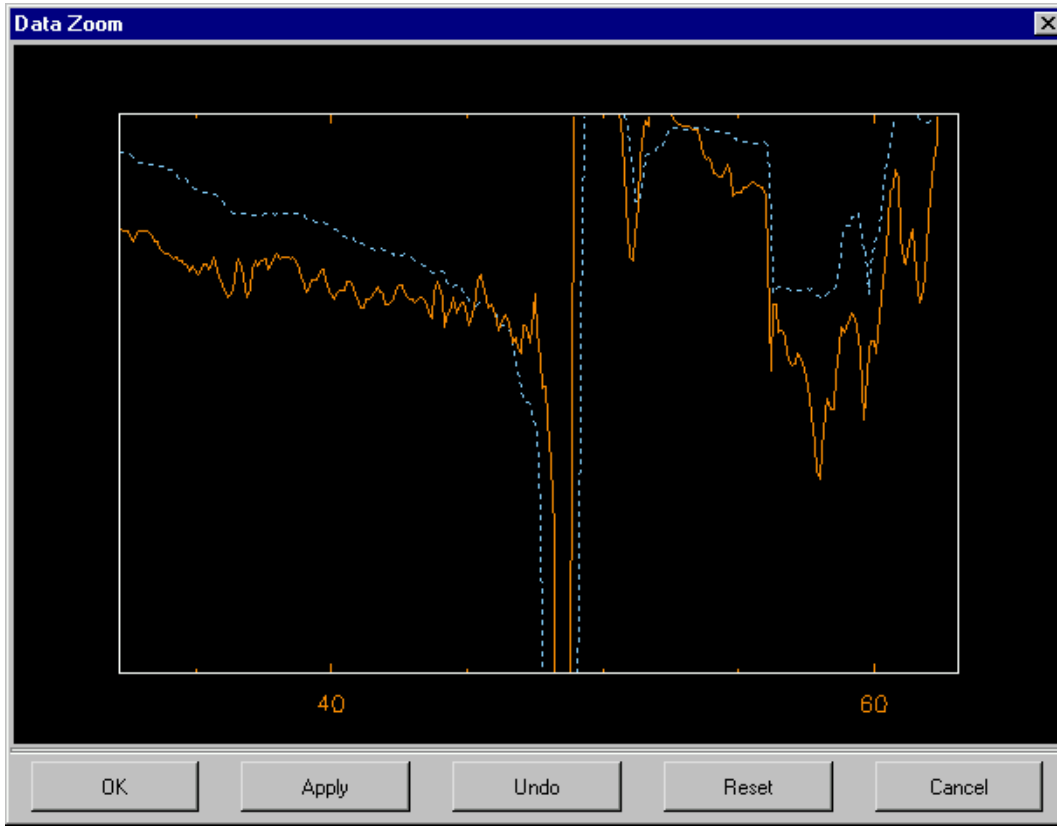


Figure 1-11 Data Zoom dialog box after zooming in on plotted data

Using Standard Functions

The Analyze menu contains built-in (standard) functions. These functions are typically used to create new, derived parameters from the data.

The standard functions include:

- **Bias** — Adds a bias value to a selected parameter and stores the result in a new parameter; the original parameter/data is unchanged.
- **Difference** — Computes the difference between two parameters and stores the result in a new parameter; the original parameter/data is unchanged.
- **Differentiate** — Computes the derivative of a selected parameter and stores the result in a new parameter; the original parameter/data is unchanged.
- **FFT** — Returns the fast Fourier transform (FFT) for the input variable. For detailed information on FFT, see the *PV-WAVE User's Guide.*
- **Gain** — Adds a gain value to a selected parameter and stores the result in a new parameter; the original parameter/data is unchanged.
- **GaussFit** — Fits a Gaussian curve through a data set; the original parameter/data is unchanged.
- **Smooth** — Smooths a selected parameter and stores the result in a new parameter; the original parameter/data is unchanged.
- **Trim** — Subtracts the initial value of a parameter from each point in the parameter and stores the result in a new parameter; the original parameter/data is unchanged.
- **WildPoint** — Removes from a parameter data points that deviate from specified limits and stores the result in a new parameter; the original parameter/data is unchanged.

For more detailed information about the functions, see *Analyze Menu* on page 82. For now, let's run one of the functions and plot its result.

Step 1 Select **Analyze=>Standard=>Smooth**. The Select Parameter to Smooth dialog box appears.

Step 2 Locate the Data Source drop-down menu and click ***src*1.**

Step 3 In the list of parameters, select the parameter **COLFEU**.

Step 4 At the top right of the dialog box locate and change the end number of points to smooth over to **3.0**.

Step 5 Click **OK**.

Step 6 Select **Edit=>Object Select** and double click in the upper graph object. The Graph Attributes Interface dialog box appears.

NOTE The parameter called **SMOOTH(*src*1:COLFEU,3)** is placed in the drop-down list and appears as 'DERIVED'. This new parameter 'DERIVED' contains the smoothed data. The original parameter, **COLFEU**, has not been changed.

Step 7 Select **AxisXY_2**.

Step 8 Select **DERIVED** from the Data Source drop-down menu.

Step 9 Click on the **SMOOTH(*src*1:COLFEU,3)**.

Step 10 Click **OK**. The derived parameter, **SMOOTH(*src*1:COLFEU,3)**, is plotted.

Using User Functions

The Analyze menu also contains built-in user-defined functions. These functions are typically used to create new, derived parameters from the data.

The user functions include:

- **rudnet_usr** — Computes the difference between parameters RPFLEU and RPFREU and stores the result in the new derived parameter.
- **dudnet_usr** — Computes the difference between parameters RPFREU and RPFLEU and stores the result in the new derived parameter.
- **get_parm_stats_usr** — Computes the statistics of a selected parameter and stores the result in a new parameter.

For more detailed information, see *Analyze Menu* on page 82. For now, let's run one of the functions.

Step 1 Select **Analyze=>User=>RudNet_usr**. The Select Source to Rudnet dialog box appears.

Step 2 Choose **src1** to create a new derived parameter.

Step 3 Click **OK**. The Select Source to Rudnet dialog box disappears.

Step 4 Select **Analyze=>User=> get_parm_stats_usr**. The Select Parameter to Obtain Statistics dialog box appears.

Step 5 Select **DERIVED** from the Data Source drop-down menu.

Step 6 Select **Rudnet_Usr(src1:RFRLEU, src1:RPFREU)** from the list of parameters. Click **OK/Apply**. Notice the statistical information appeared in the right panel widow.

Step 7 Click **Cancel** to exit the window.

NOTE A new parameter **Rudnet_Usr(src1:RFRLEU, src1:RPFREU)** has been added to source id 'DERIVED' list of parameters. The new parameter contains the difference between parameters RPFLEU and RPFREU data. The original parameters are unchanged.

Moving and Aligning Graphs

Step 1 Select **File=>Page Setup**. Select both the **Landscape** button and **Tabloid** from the list of available sizes**.**

Step 2 Click **OK**. The drawing area is now larger and oriented horizontally. Now you can reposition the header and the two graphs.

Step 3 Select **Edit=>Select All**.

Step 4 Position the mouse pointer inside the upper graph, hold down the left mouse button and drag the graph to the right, until all three *y*-axes are visible.

Step 5 Use the mouse again to drag the header to the right so that it is positioned above the graph.

Now, we will use the **Align Graphs** function to automatically line up the lower graph with the upper graph.

Step 6 Select **Edit=>Deselect All**.

Step 7 Click in the upper graph to select it.

Step 8 Select **Edit=>Align Graphs**.

The lower graph is automatically aligned with the upper graph. The **Align Graphs** function automatically aligns all other graphs in the drawing area with the selected graph. In addition, the *x*-axes for all graphs are scaled to match the *x*-axis of the selected graph.

Creating a Tabular Data File

The **Create TabData** function allows you to write a subset of the currently loaded data file into a tabularly arranged ASCII file. Tabular files consist of header information and columns of data. The tabular data file can be used for reports or read back into the TS-WAVE for further analysis. (If a tabular file is read back into TS-WAVE, the headers are ignored.)

Figure 1-12 shows part of a tabular data file generated by TS-WAVE.

```
Created: 10/11/2001 13:35:06.000    C:\VNI\tswave-3_0\tab\t.tab
 Data Source: C:\VNI\tswave-3_0\data\data.ldf
 Start: 05/17/1996 08:31:09.859  Stop: 05/17/1996 08:44:26.506
 Identifier: N/A       Sample Rate: N/A
 Long :
 Units:
 Param:                  ACCG   AOSNEU
05/17/1996 08:31:19.557   184.763   05/17/1996 08:31:19.557   0.614937
05/17/1996 08:31:19.656   184.763   05/17/1996 08:31:19.656   0.687748
05/17/1996 08:31:19.756   184.763   05/17/1996 08:31:19.756   0.687748
05/17/1996 08:31:19.855   184.763   05/17/1996 08:31:19.855   0.687748
05/17/1996 08:31:19.955   184.763   05/17/1996 08:31:19.955   0.760567
05/17/1996 08:31:20.057   184.763   05/17/1996 08:31:20.057   0.687748
05/17/1996 08:31:20.156   184.763   05/17/1996 08:31:20.156   0.760567
05/17/1996 08:31:20.256   184.763   05/17/1996 08:31:20.256   0.760567
05/17/1996 08:31:20.355   184.763   05/17/1996 08:31:20.355   0.760567
05/17/1996 08:31:20.455   184.763   05/17/1996 08:31:20.455   0.760567
05/17/1996 08:31:20.557   184.763   05/17/1996 08:31:20.557   0.760567
05/17/1996 08:31:20.656   184.763   05/17/1996 08:31:20.656   0.760567
05/17/1996 08:31:20.756   184.763   05/17/1996 08:31:20.756   0.760567
05/17/1996 08:31:20.855   184.763   05/17/1996 08:31:20.855   0.833401
05/17/1996 08:31:20.957   184.763   05/17/1996 08:31:20.957   0.833401
05/17/1996 08:31:21.057   184.763   05/17/1996 08:31:21.057   0.833401
05/17/1996 08:31:21.156   184.763   05/17/1996 08:31:21.156   0.833401
05/17/1996 08:31:21.256   184.763   05/17/1996 08:31:21.256   0.833401
05/17/1996 08:31:21.355   184.763   05/17/1996 08:31:21.355   0.906246
05/17/1996 08:31:21.455   184.763   05/17/1996 08:31:21.455   0.906246
05/17/1996 08:31:21.555   184.763   05/17/1996 08:31:21.555   0.906246
05/17/1996 08:31:21.654   184.763   05/17/1996 08:31:21.654   0.906246
05/17/1996 08:31:21.756   184.763   05/17/1996 08:31:21.756   0.833401
05/17/1996 08:31:21.855   184.763   05/17/1996 08:31:21.855   0.906246
05/17/1996 08:31:21.955   184.763   05/17/1996 08:31:21.955   0.906246
```

Figure 1-12 A tabular data file generated by TS-WAVE.

Step 1 Select **Create=>Create TabData**. This brings up a standard file creation dialog box.

Step 2 In the dialog box, specify a filename and directory for the tabular data file and click **OK**. This brings up the Select Source ID dialog box, shown in *Figure 1-13*.

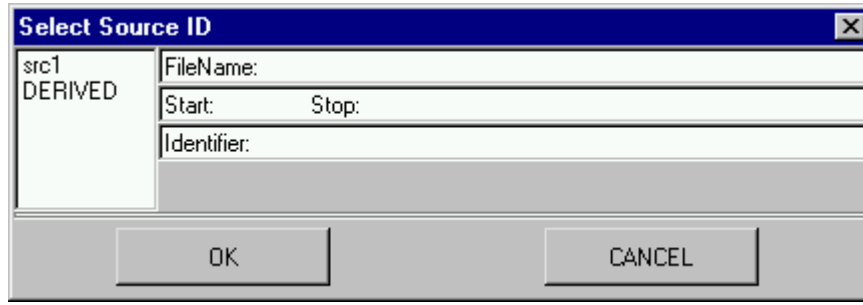


Figure 1-13 The Select Source ID dialog box

Step 3 In the Select Source list, select **src1**. Notice the Start and Stop time of the Data Source is displayed. Click **OK**. This brings up the Select Tab Attributes dialog box as shown in *Figure 1-14*

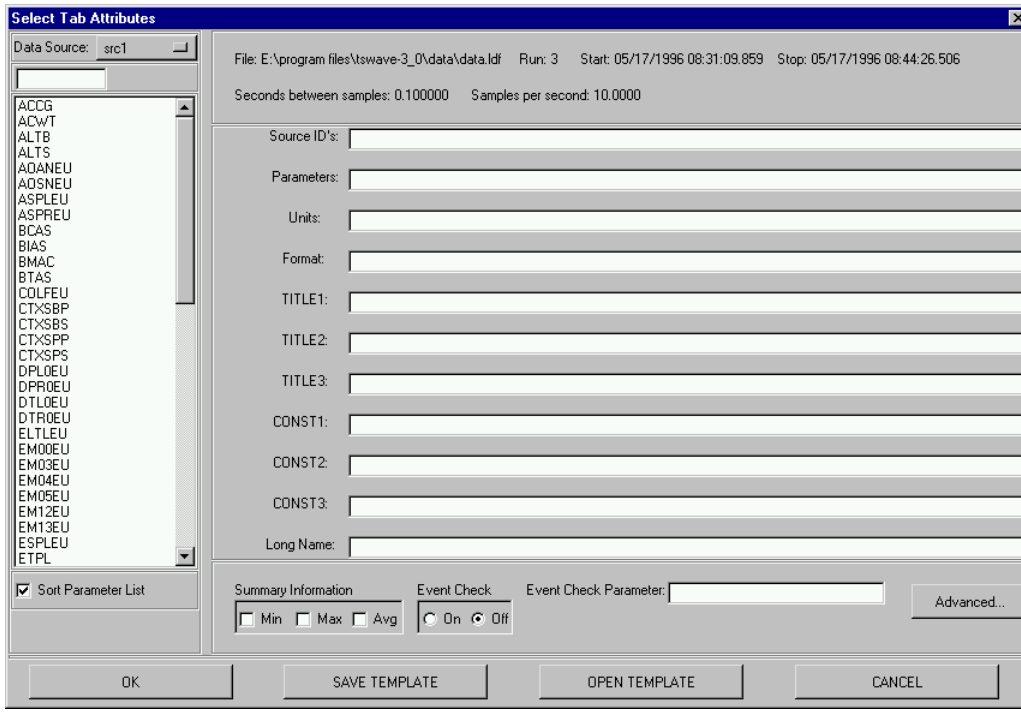


Figure 1-14 The Select Tab Attributes dialog box lets you specify the parameters and other information to write to a tabular data file.

Step 4 In the parameter list, click the following parameter names: **COLFEU**, **ESPLEU**, and **ALTB**. When you click a parameter, its name is added to the **Parameters** text field. When you double click a parameter, its name is added to the **Parameters** text field and its code is formatted, and its statistics are displayed in the **Format** text field.

Step 5 Enter the names of the parameters in the **Title 1**, **Title 2**, and **Title 3** fields. These parameters are Elevator Force, Elevator Position, and Boom Altitude.

Step 6 In the **Long Name** field, enter the long names for the parameters. Separate words with an underscore and separate names with a space. For example:

```
Elevator_Force Elevator_Position Boom_Altitude
```

Step 7 Select the **Min**, **Max**, and **Avg** buttons.

Step 8 Click **OK** to create the tabular data file. For now, don't delete this file, because you will use it later in the tutorial.

Step 9 View the tab data file in any text editor, or use the **Create=>View Tab-Data** function. The editor used by TS-WAVE is defined in the file *VNI_DIR*/tswave-3_0/resource/tswave_files.ads. The default for Windows is Notepad; the default for UNIX is vi.

TIP If you wish to reuse your tab attributes settings in the future, create a tab attributes template.

To create a tab attributes template, click **SAVE TEMPLATE** (in the Select Tab Attributes dialog box) and specify a template name and directory.

If you want to reuse this template, simply select the **OPEN TEMPLATE** button to load the template back in to the Select Tab Attributes dialog box.

Saving and Restoring a Session

Use the **File=>Save Session** command to save an entire session, including the currently loaded/displayed data. When you restore a session (with **File=>Open Session**) the session is restored exactly as it was when it was saved.

Creating a graph on XY parameters

Select **File=>New** to open a new page to create a graph on XY parameters. Use the **Create=>Graph Object** function to create an XY Plot in the drawing area. If you wish, add a header to the drawing area and name the plot XY Plot of Parameters.

Loading a Tab Data File and Choosing Parameters

Step 1 Select **File=>Open**. Notice the Open Data File dialog box appears as shown in *Figure 1-2*.

Step 2 Select the `tswave-3_0/tab` directory from the Open Data File dialog box and then pick a tabular data file and click **OK**. Notice the Name and Type of Data Source dialog box appears.

Step 3 Select **src3** from the Name and Type of Data Source dialog box and then click **OK**.

Step 4 Select **Edit=>Object Select**.

Step 5 Double click on the graph. The Graph Attributes Interface dialog box appears as shown in *Figure 1-8* on page 16. Select **src3** from the Data Source drop-down menu, then select **COLFEU**, the *y*-axis variable (the dependent variable), from the Parameter list and click **APPLY** to update the graph.

Step 6 Click on the X Axis tab on the far upper right of the Graph Attributes Interface. Notice that the list of parameters is grayed out. Click on the Time Series Plot box near the bottom left of the interface. This option determines the nature of the X Axis data. The default is to plot a graph object in a Time Series mode. By clicking on the Time Series Plot box, you have just enabled the list of parameters to put TS**-**WAVE into the X-Y plot mode. Select **src3** from the Data Source drop-down menu, then select the parameter **ALTB**, and click **APPLY** to update the graph. *Figure 1-15* displays the Graph Attributes Interface dialog box with the X Axis tab activated.

NOTE The parameter's name is case sensitive.

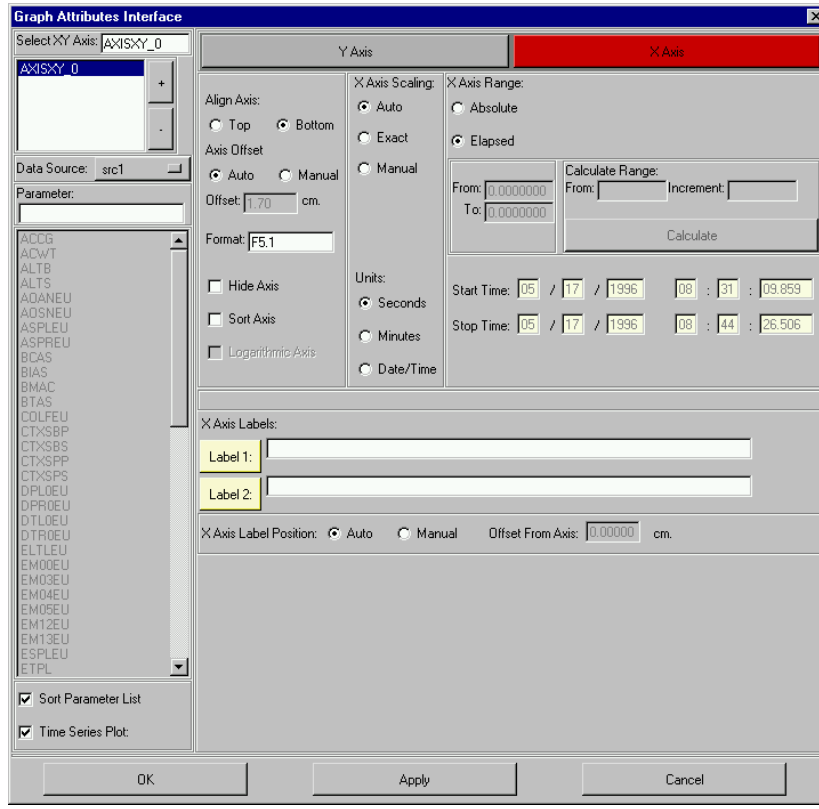


Figure 1-15 Graph Attributes dialog box with X Axis tab activated

Step 7 In the X Axis Labels, Label 2 text field, enter a name for this parameter (`Boom Altitude`).

Step 8 Click **OK**.

The specified variables are plotted. You can plot additional *y*-axis variables by addding the additional **XY Axis** setting, as you did in the previous exercise.

Congratulations, you have completed the TS-WAVE tutorial.

Where to Find More Information

For information on specific TS-WAVE menu functions and dialog boxes, refer to Chapter 3, *GUI Reference*.

See the following section for information on additional TS-WAVE topics.

Additional Topics

This section discusses the following topics:

- creating a pick file
- saving and using a template
- creating a batch file

Creating a Pick File

A pick file is simply a tabular data file that contains data points taken at selected time slices or at a specified range of times.

NOTE For this procedure, you can use the same tab data file that you created previously in the tutorial in the section *Creating a Tabular Data File* on page 26.

Step 1 Select **Create=>Open Pick File**.

Step 2 In the Pick File dialog box, enter a name for the tabular data file. Click **OK**. This brings up the Select Source ID dialog box.

Step 3 In the Select Source list, select src1. Notice the Start and Stop time of the Data Source is displayed. Click **OK**. The Select Tab Attributes dialog box appears with the currently plotted parameters already selected.

Step 4 If you wish, use the Select Tab Attributes dialog to add data titles, names of constants, summary information, and other information and formatting features to the tabular (pick) file. See *Select Tab Attributes Dialog Box* on page 89 for detailed information on this dialog box. Click **OK** when you are finished selecting tab attributes.

Step 5 In the drawing area, a vertical line appears. Use the mouse to move this line across the plot. **Click** and **drag** your left mouse button to select as many different time slices as you wish. Only the data for those parameters defined in the Select Tab Attributes dialog box that fall on the time slices (on the vertical line) are written to the tabular data file.

Step 6 Select **Create=>Close Pick File** to close and save the tabular data file.

Step 7 If you wish, use **Create=>View Tab Data** to examine the file.

Saving and Using a Template File

Normally, test engineers wish to look at the same data parameters collected over many different time sectors. TS-WAVE lets you set up graphs for specific parameters and save the entire setup as a template, which can be reused with data from different runs or flights.

For instance, you might create a plot that contains three graphs that plot eight different parameters. A template of this session contains the same graphs, text, headers, and other graphics objects, and the same selected data parameters. In addition, information is saved to recompute any derived parameters that were created with Standard or User Defined functions.

All you have to do is open the template (select **File=>Open Tempate**) and load the new data. The new data is automatically plotted.

To save a template, follow these steps. We assume that you have already created a session and plotted some data parameters.

Step 1 Select **File=>Save Template**.

Step 2 Specify a filename and directory in the Save Template As dialog box.

To restore a previously saved template, do the following:

Step 1 Select **File=>Open Template**. The Restore Template dialog box appears.

Step 2 Select a template. Click **OK**. The “Template Source Association dialog box appears.

Step 3 Select an existing Data Source ID from the list and click **OK**. The selected Data Source ID is now assigned to the Templates Source ID. The new Data Source ID appears in the list of available Data Sources. Click **OK** to exit. You can also select a Data Source ID by pressing the Browse button.

NOTE If you choose to select a Data Source ID via the Browse button you are skipped through the **File=>Open** process. If the data source id you selected is valid it appears highlighted in the list of available Source IDs. If the selection is not valid, a message box appears. Click **OK** to continue.

Step 4 If the template uses more than one Source ID, step 3 is automatically repeated until all Source IDs within the template are assigned. After all

of the Data Source IDs have been assigned, the template is displayed with all of the assigned Data Source IDs listed.

NOTE If parameters in the assigned data source do not match the parameters in the template a message box appears. To change the parameters, double click the graphic object to open the Graph Attributes Interface dialog box.

NOTE Templates are also used to create batch files, as described in the next section.

Creating a Batch File

To process your data in batch mode, you need to create a batch file. A batch file allows you to process numerous flights or runs without user interaction.

Batch Mode Output

The TS-WAVE Batch command produces several files, not just a single batch file. The first file is a file that you name, and that has a `.job` extension. This file contains a list of data files and associated TS-WAVE templates. The batch file (the Windows version) is structured as follows:

```
BEGIN
    PRINTER      =PS3
    PRINT_DRIVER =PS
    PRINT_TO_FILE=<Not Set>

    Time_Start=05/17/1996 08:31:09.859
    Time_Stop =05/17/1996 08:44:26.506
    Source=src1
    Run=0
    Desample=90
    Data_Source=C:\VNI\tswave-3_0\data\data.ldf
    DataHandler=ldf

    Template=C:\VNI\tswave-3_0\batch\\a_1.tpl
END
```

A command file is created automatically, with an extension `.bat`. This file is placed in the same directory as the `.job` file. The command file contains the following information:

- the main TS-WAVE installation directory
- an TS-WAVE command, which invokes TS-WAVE in batch mode with the specified batch file as input
- an exit command

For example:

```
cd C:\VNI\tswave-3_0

..\bin\bin.i386nt\tswave -j C:\VNI\tswave
3_0\batch\my.job
exit
```

Template files are also saved in the batch file directory. These files have a `.tpl` extension, such as `mybatch_2.tpl`, and so on.

When a batch file is run, the following steps occur:

Step 1 A TS-WAVE session starts in the background.

Step 2 The first template is opened.

Step 3 The template's associated data files are loaded.

Step 4 Steps 2, and 3 are repeated until all templates in the batch file have been processed.

Step 5 Batch mode is exited.

The graphs generated by the batch file are automatically sent to the default printer. Typically, TS-WAVE batch files are set up to execute automatically using a job scheduler or timer.

Saving a Batch File

This procedure assumes that a *data.ldf* file is loaded and is assigned to src1; a graph object exists, and at least one parameter is plotted in a graph object. The procedure for creating a batch file is as follows:

Step 1 Select **Create=>Open Batch File**.

Step 2 In the Open Batch File dialog box, enter a filename and specify a directory for the batch file. Click **OK**.

Step 3 Select **Create=>Add Template to Batch** to add the current drawing area configuration (graphs, headers, data, and so on) to the batch file.

Step 4 Select **File=>Open.** The Open Data File dialog box as shown in *Figure 1-2*.

Step 5 Select *data.ldf* and click **OK**. The Name and Data Source dialog box appears as shown in *Figure 1-3*.

Step 6 Enter **src1** in the Source ID field.

Step 7 Select **ldf** format.

Step 8 Click **OK**. The Select Data Run dialog box appears.

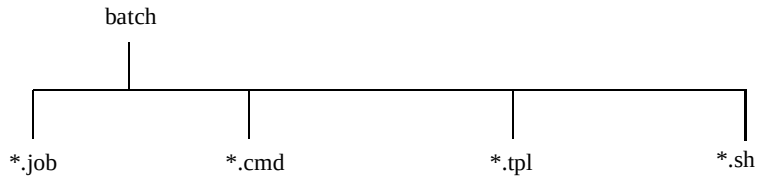
Step 9 Select a run you would like to add to the batch file.

Step 10 Click **OK**. The selected run will be processed and plotted using the template when the batch file is run.

Step 11 Repeat steps 4-10 to associate additional templates and data files in the batch file.

Step 12 Select **Create=>Close Batch File** to close and save the batch file.

The batch file, command file, shell script, and template files are saved in the same directory. The shell script (`*.sh file`) is only created on UNIX systems. For example:



When a batch file is executed, graphs are generated and automatically sent to the default printer specified in the resource file:

```
$TSWAVE_ROOT/resources/tswave_print.ads
```

For detailed information on resource files, see *Resource Files* on page 111.

NOTE If the file is not set up for your printer, you will need to edit the files `batch_name.sh` and `batch_name.job` to point to your printer.

Executing a Batch File on Windows Systems

After you select **Close Batch File**, you can run the batch file by executing the `.bat` file that the batch command created. You can execute the command file by double clicking on it in Windows or Windows NT Explorer, or using an automated job scheduler.

Executing a Batch File on UNIX Systems

After you select **Close Batch File**, you can run the batch file by running the Bourne shell script (`*.sh` file) that the batch command created.

To execute the shell script under a Bourne or Korn shell, enter the following command at the shell prompt:

```
batch_name.sh 2>&1 output_message_file &
```

To execute the shell script under a C shell, enter following command at the shell prompt:

```
batch_name.sh >& output_message_file &
```

You can also execute the shell script as part of a `cron` job, or other process scheduler.

2

CHAPTER

Developer's User Reference

The Developer's User Reference contains the following sections:

- *TS-WAVE User Functions* — TS-WAVE's User Functions, User Function Requirements, Writing a User Function, Modifying User Function Directory, Creating a Derived Parameter, Creating a Help File, Creating a New Plot Window, A Complete UserFCN Example, and Tips for Creating User Functions
- *TS-WAVE DataHandlers* — TS-WAVE's DataHandlers, DataHandler Requirements, and Creating a DataHandler
- *TS-WAVE Utility Routines* — TS-WAVE's DataManager Routines, and TS-WAVE's User Interface Routines

TS-WAVE User Functions

General

In addition to TS-WAVE's standard analysis functions, TS-WAVE provides a mechanism for adding a wide range of customized functionality. User functions can be added easily to integrate analysis routines that call an external C or FORTRAN library, generate a derived parameter, or display a custom plot.

User Function Requirements

There are a number of requirements for a TS-WAVE user function. If any of these requirements are not met, the functionality will not be usable from within TS-WAVE.

A user function must have the following:

- Be a PV-WAVE procedure. This could include: other PV-WAVE routines, calls to an external C or FORTRAN library, graphical output creation, read and write files, and connection to a database.
- Be located in the `UserFunction_Root_Dir` directory defined in *tswave.ads*. The default for this directory is `tswave_dir/usrfcn`. The plain text resource file *tswave.ads* is located in the `tswave_dir/resource` directory.
- Have a name that ends in "*_usr*". For example, *youruserfcn_usr* in the file `youruserfcn_usr.pro`.
- Accept one parameter, a string containing the unique VDA tool name.
- Be listed in the file `tswave_dir/usrfcn/userfunction.list`. This will add the function to the **Analyze =>User** menu.

Writing a User Function

Follow PV-WAVE's VDA tool procedures when writing a user function. For instance, use PV-WAVE's Wo* and Ww* functions when creating user interfaces. For details, see the Application Developer's Guide. There are special rules for creating derived parameters, for details see, Creating a Derived Parameter *on page 42*. Additionally, tools are provided to aid you in building a user interface that interacts with open data source files and their available parameters. These tools are discussed throughout this section.

A thorough understanding of how to use TS-WAVE's utilities to access source ID names, parameter names, and data will aid in your development process. See *TS-WAVE Utility Routines* on page 64.

The steps below describe how to write a user function.

1. Create a PV-WAVE procedure and name it *yourusrfcn_usr.*

NOTE We will use *yourusrfcn* throughout the remaining steps and examples.

---

2. Accept one positional parameter, a string containing the unique VDA Tool name.

Example

Here is a simple example of a user function we named *yourusrfcn_usr.*

```
PRO   yourusrfcn_usr, tool_name
      ...
END
```

3. Add customized code to perform a specific task. When the specific task is done and *yourusrfcn_usr* is completed, go to Step 4.

4. Add *yourusrfcn_usr* to the UserFunction list, which is located in `TSWAVE_DIR/usrfcn/UserFunction.list`.

Example

Here is an example of a *UserFunction.list*.

```
; comment line
;
rudnet_usr
dudnet_usr
get_parm_stats_usr
yourusrfcn_usr
```

Modifying the UserFCN Directory

If you wish to store your *userfcn* in a directory other than the default directory, you must modify the `tswave.ads` subdirectory.

In the example below, which was extracted from *tswave.ads*, `UserFunction_Root_Dir` is not set. `UserFunction_Root_Dir` is set as the default, `TS-WAVE_DIR`. `UserFunction_File_Dir` is set to `usrfcn`. `DH_TypeListFile` is set to the default filename *UserFunction.list.* When TS-WAVE starts it searches in `TS-WAVE_DIR/UserFunction/userfunction.list` for all user functions and registers them with the application. `UserFunction_HelpFileExt` prompts TS-WAVE to search for the User Function Help files with *.txt* extension.

Example

```
Extracted from TSWAVE_DIR\resource\tswave.ads
;******************
!-----------------------------------------
!TsWave User Function Files, etc
!Leave directory blank to use the default under TS-WAVE's;
   home directory.
!-----------------------------------------------
UserFunction_Version: 3.0
UserFunction_Root_Dir:
UserFunction_File_Dir:     usrfcn
UserFunction_ListFile:     userfunction.list
UserFunction_HelpFileExt:  txt
;***********************
```

Creating a Derived Parameter

A TS-WAVE user function is commonly used to create a new variable, either as an intermediate or a final result to allow further analysis or to output graphically. This section describes how easy it is to create a derived parameter.

To allow TS-WAVE to access the new parameter after the user function has gone out of scope, DM_addFcn must be called within the *youruserfcn_usr* function. For more details on DM_addFcn function, see *DM_addFCN Function* on page 64. A separate function must be created that is called when the derived parameter is used. This function can reside in the same *.pro* file as the *youruserfcn_usr* function, or, if you wish, it can be created as a separate file in the `tswave_dir/usrfcn` directory.

Here's the method to use to call *youruserfcn.*

```
data = youruserfcn( tool_name, src_name, parm_name)
```

Where the `src_name` and `parm_name` parameters are obtained as shown in the following code:

```
ret_data = GU_readParmlist_component( tool_name, wids=parm_wids )
parm_name = ret_data('PARAMETER')
src_name = ret_data('SOURCE')
```

This function, *youruserfcn*, must return an associative array with a key named DATA, which contains an array of data values, and a key named TIME, a PV-WAVE Date/Time structure (!DT). If the parameter does not have a time array, set the values in the TIME key to an indexed array of integers with the same number of elements as the data array: `INDGEN(N_ELEMENTS(datavalues)-1)`.

Example

Here is a simple example of *youruserfcn_usr* function. The data array is squared and the time array is passed back unchanged.

```
FUNCTION UDemo, tool_name, src_name, parm_name
    data_array = DM_getParm( tool_name, src_name, parm_name )
    IF  size(data_array,/type) NE 11  THEN BEGIN
        Print, 'DM_getParm, did not return data'
        RETURN, ASARR()
    ENDIF
    data = data_array('DATA')
    time = data_array('TIME')

    ; do some simple math
    data2 = data*data
    RETURN, ASARR('DATA', data2, 'TIME', time)
END
```

Creating a Help File

You may include a help file for your user function. The help file must be a plain text file and have the same name as the function, but have a '*txt*' extension. Additionally, the file must be located in the same directory as the user function. Users will be able to access this help file by selecting **Help =>User Fcn Help** on the menu bar.

Creating a New Plot Window

If a user function contains a WwDrawing window, it must manage the new window id.

Example

This example uses the WSET procedure.

```
wid = WwDrawing (layout, windowid, $
                 'VzZoomToolExposeCB', $
                 wsize, dsize, $
                 /NoMeta, /Noscroll, $
                 Area = area, $
                 /Top, /Bottom, /Left, /Right)
status = TmSetAttribute(tool_name, 'new', 'WINDOWID', windowid)
status = WwSetValue(area, UserData=tool_name)

; Set up a handler for button presses (a zoom).  When the
; mouse is clicked, motion and release handlers are initiated.
status = WwHandler(area, 'VzZoomPressHandler', $
                   'ButtonPressMask', area)
  …
  …
  …
windowid = TmGetAttribute(tool_name, 'new', 'WINDOWID', $
                          Default=-1)
IF windowid EQ -1 THEN RETURN
WSET, windowid
```

A Complete User Function (userfcn) Example

This example creates the following:

- Graphical user interface that contains a parameter list
- Text box that contains statistics derived from the selected parameter
- Drawing window with which one can interact with the mouse, and also creates a derived parameter.

To test this example, do the following:

Step 1 Copy the text below into a plain text file called *udemo_usr.pro* and place it in `tswave_dir/usrfcn`, then update the `tswave_dir/usrfcn/userfunction.list` file as discussed in *Writing a User Function* on page 40.

Step 2 Start TS-WAVE and load a data set, then select **Analyze => User => udemo_usr** from the menu bar.

Step 3 Select a parameter from the list and click **OK**. Notice a time series plot appeared in the drawing window. **Click** and **drag** your left mouse button to select start and end points. The statistics in the box are updated and the selected time series segment is highlighted. Right click to return to the original plot. To access the derived parameter, click on the **Data Source** pull-down menu and choose **DERIVED** as the data source. To close the window, select **Cancel**.

Example

```
;*****************************************************************
   *************
; Drawing Callback
;*****************************************************************
   *************
PRO udemo_tsdrawCB, wid, index
    COMMON stuff, tool_name, parm_wids, flag, x0, x1, y0, y1
    COMMON other_stuff, data, textwid, parm_name
    DECLARE FUNC, TmGetAttribute
    DECLARE FUNC, TmSetAttribute
    DECLARE FUNC, DM_getParm
    DECLARE FUNC, GU_read_Parmlist_component
    DECLARE FUNC, UDemo

    IF SIZE(flag, /Type) EQ 0 THEN flag = 0

    DPRINT, 'In udemo-tsDrawCB...'
    SetCursor, tool_name, /Wait

    IF SIZE(parm_wids, /Type) LT 8 THEN RETURN
```

```
    ret_data = GU_readParmlist_component( tool_name, wids=parm_wids )
     parm_name = ret_data('PARAMETER')
     src_name  = ret_data('SOURCE')

    textwid = TmGetAttribute( tool_name,'PARM_STATS_WIDS','TEXT_WID'
    )
     data_array = DM_getParm( tool_name, src_name, parm_name)

     IF  SIZE(data_array,/type) NE 11 THEN BEGIN
         MESSAGE, 'DM_getParm, did not return data', /Continue
         RETURN
     ENDIF

     data = data_array('DATA')
     time = data_array('TIME')

     ; call UDemo() to compute y
     yc = UDemo(tool_name,src_name,parm_name)

     ; pull back the computed y values
     y = yc("DATA")

    ; add y as a new parameter and update the list (appears in DERIVED
    source)
     newparm = 'UDemo_Usr(' + STRTRIM(src_name(0),2) + '!' +
    parm_name(0) + ')'
     exe_cmd = "UDemo(tool_name,'" + src_name + "','" + parm_name +
    "')"
     rv = DM_addFCN(tool_name, newparm, exe_cmd)

     ; plot the new parameter
     PLOT, data, title = STRSUBST(parm_name,'!','!!')
     SetCursor, tool_name, /Reset
END
```

```
;*****************************************************************
   *************
; Event Handler Callback
;*****************************************************************
   *************
PRO UDemo_handlerCB, wid, shell, event
    COMMON stuff, tool_name, parm_wids, flag, x0, x1, y0, y1
    COMMON other_stuff, data, textwid, parm_name
    DECLARE FUNC, TmGetAttribute
    DECLARE FUNC, TmSetAttribute
    DECLARE FUNC, DM_getParm
    DECLARE FUNC, GU_read_Parmlist_component
    DECLARE FUNC, Display_Parm_StatsCB

    IF event.button EQ 1 THEN BEGIN     ; Left Mouse Button
        xypos = WwGetPosition( event )

        xdev = xypos(0)
        ydev = !D.Y_Vsize - xypos(1)

        xydat = CONVERT_COORD(xdev, ydev, /Device, /To_Data)

        test = flag MOD 2
        IF (test EQ 0) THEN BEGIN
            x0 = xydat(0)
            y0 = xydat(1)
            XYOUTS, x0, xydat(1), '*', color='FFFF00'xl
        ENDIF ELSE BEGIN

            x1 = xydat(0)
            y1 = xydat(1)
            XYOUTS, x1, xydat(1), '*', color='FF00FF'xl
```

```
        PLOTS, [x0,x1], [y0,y1], color=5

        tmp = [x0,x1]    &   tmp = tmp(SORT(tmp))
        x0 = tmp(0)      &   x1 = tmp(1)
        datarange = data(FIX(x0):FIX(x1))
        OPLOT, FINDGEN(x1-x0)+x0-1, datarange, color=16
    ENDELSE
    flag = flag + 1
ENDIF ELSE BEGIN                    ; Not Left Mouse Button
    flag = 0
    test = 1
    PLOT, data, title = STRSUBST(parm_name,'!','!!')
    datarange = data
ENDELSE

IF (test NE 0) THEN BEGIN   ; Update statistics
        parmsize = N_ELEMENTS(datarange)
        strparmsize = STRING(REPLICATE(32B,40))
        STRPUT,strparmsize,'Size of Parameter:'
      STRPUT,strparmsize,STRCOMPRESS(STRTRIM(STRING(parmsize),
2)),29

        DH_MinMax, datarange, Max=maxval, Min=minval
        strminval = STRING(REPLICATE(32B,40))
        STRPUT,strminval ,'Minimum Value:'
        STRPUT,strminval , STRCOMPRESS(STRTRIM(STRING(minval),
2)),29

        strmaxval = STRING(REPLICATE(32B,40))
        STRPUT,strmaxval,'Maximum Value:'
        STRPUT,strmaxval, STRCOMPRESS(STRTRIM(STRING(maxval),
2)),29

        meanval = MEDIAN(datarange)
        strmeanval = STRING(REPLICATE(32B,40))
```

```
            STRPUT,strmeanval,'Median Value:'
           STRPUT,strmeanval, STRCOMPRESS(STRTRIM(STRING(meanval),
    2)),29

            stdval = STDEV(datarange)
            strstdval = STRING(REPLICATE(32B,40))
            STRPUT,strstdval ,'Standard Deviation:'
           STRPUT,strstdval , STRCOMPRESS(STRTRIM(STRING(stdval),
    2)),29

            totval = TOTAL(datarange)
            strtotval = STRING(REPLICATE(32B,40))
            STRPUT,strtotval,'Total of Parameter:'
           STRPUT,strtotval , STRCOMPRESS(STRTRIM(STRING(totval),
    2)),29

            rangeval = maxval - minval
            strrangeval = STRING(REPLICATE(32B,40))
            STRPUT,strrangeval,'Range of Parameter:'
         STRPUT,strrangeval, STRCOMPRESS(STRTRIM(STRING(rangeval),
    2)),29

            strret = STRING(10B)
            strtxt = strparmsize + strret + strminval + strret +
    strmaxval + $
               strret + strmeanval + strret + strstdval + strret + $
                 strtotval + strret + strrangeval + strret
            st = WwSetValue( textwid , strtxt )
     ENDIF
END

;*****************************************************************
    *************
; Display Stats Callback
;*****************************************************************
    *************
```

```
FUNCTION Display_Parm_StatsCB, wid, index
    COMMON stuff, tool_name, parm_wids, flag, x0, x1, y0, y1
    DECLARE FUNC, TmGetAttribute
    DECLARE FUNC, TmSetAttribute
    DECLARE FUNC, DM_getParm
    DECLARE FUNC, GU_read_Parmlist_component

    DPRINT, 'In Display_Parm_StatsCB...', index
    SetCursor, tool_name, /Wait
    IF(index EQ 3) THEN return,0

    udata  = WwGetValue( wid, /Userdata )
    tool_name = udata(0)
    parm_wids = udata(1)
    grael_name = 'GET_PARM_STATS_USR'

   ret_data = GU_readParmlist_component( tool_name, wids=parm_wids )
    parm_name = ret_data('PARAMETER')
    src_name  = ret_data('SOURCE')

   textwid = TmGetAttribute( tool_name,'PARM_STATS_WIDS','TEXT_WID'
    )

    data_array = DM_getParm( tool_name, src_name, parm_name)

    IF  SIZE(data_array,/type) NE 11 THEN BEGIN
        PRINT, 'DM_getParm, did not return data'
        RETURN, ASARR()
    ENDIF

    data = data_array('DATA')
    time = data_array('TIME')

    ; Do the desired statistics
```

```
parmsize = N_ELEMENTS(data)
strparmsize = STRING(REPLICATE(32B,40))
STRPUT,strparmsize,'Size of Parameter:'
STRPUT,strparmsize,STRCOMPRESS(STRTRIM(STRING(parmsize), 2)),29

DH_MinMax, data, Max=maxval, Min=minval
strminval = STRING(REPLICATE(32B,40))
STRPUT,strminval ,'Minimum Value:'
STRPUT,strminval , STRCOMPRESS(STRTRIM(STRING(minval), 2)),29

strmaxval = STRING(REPLICATE(32B,40))
STRPUT,strmaxval,'Maximum Value:'
STRPUT,strmaxval, STRCOMPRESS(STRTRIM(STRING(maxval), 2)),29

meanval = MEDIAN(data)
strmeanval = STRING(REPLICATE(32B,40))
STRPUT,strmeanval,'Median Value:'
STRPUT,strmeanval, STRCOMPRESS(STRTRIM(STRING(meanval), 2)),29

stdval = STDEV(data)
strstdval = STRING(REPLICATE(32B,40))
STRPUT,strstdval ,'Standard Deviation:'
STRPUT,strstdval , STRCOMPRESS(STRTRIM(STRING(stdval), 2)),29

totval = TOTAL(data)
strtotval = STRING(REPLICATE(32B,40))
STRPUT,strtotval,'Total of Parameter:'
STRPUT,strtotval , STRCOMPRESS(STRTRIM(STRING(totval), 2)),29

rangeval = maxval - minval
strrangeval = STRING(REPLICATE(32B,40))
STRPUT,strrangeval,'Range of Parameter:'
STRPUT,strrangeval, STRCOMPRESS(STRTRIM(STRING(rangeval), 2)),29
```

```
    strret = STRING(10B)
    strtxt = strparmsize + strret + strminval + strret + strmaxval
   + strret + $
       strmeanval + strret + strstdval + strret + strtotval + strret
   + $
        strrangeval + strret
    st = WwSetValue( textwid , strtxt )

    UDemo_TSDrawCB, wid, index
    SetCursor, tool_name, /Reset
    RETURN, 1
END


;*****************************************************************
   *************
; Derived param function
;*****************************************************************
   *************
FUNCTION UDemo, tool_name, src_name, parm_name

    data_array = DM_getParm( tool_name, src_name, parm_name )

    IF  size(data_array,/type) NE 11  THEN BEGIN
        PRINT, 'DM_getParm, did not return data'
        RETURN, ASARR()
    ENDIF

    data = data_array('DATA')
    time = data_array('TIME')

    ; do some math
    data2 = data*data

    ; return data, time and anova_table in an As.Arr
```

```
    RETURN, ASARR('DATA', data2, 'TIME', time)
END

;*****************************************************************
   *************
; User Function UDemo_Usr
;*****************************************************************
   *************
PRO UDemo_Usr, tool_name
    ; Forward declaration of TM routines
    ; ----------------------------------
    DECLARE FUNC, TmGetAttribute
    DECLARE FUNC, TmSetAttribute
    DECLARE FUNC, TmGetTop
    DECLARE FUNC, WoGenericDialog
    DECLARE FUNC, GetToolName
    DECLARE FUNC, GU_newParmlist_component
    DECLARE FUNC, GU_read_Parmlist_component
    DECLARE FUNC, GU_updateParmlist_component

    buttons = LONARR(3)
    dialog = WoGenericDialog( TmGetTop( tool_name ),    $
                layout,                                 $
                'Display_Parm_StatsCB',                 $
                Title = 'Select Parameter',             $
                /OK, /Apply, /Cancel, /NoDestroy,       $
                Default = 1, Buttons = buttons          )

    Status = TmSetAttribute( tool_name, 'DIALOG_IDS', $
                             'GET_PARM_STATS_USR_GUI', dialog )

    ; layout of widgets
    llay = WwLayout( layout, /Top, /Left, /Bottom, /Form, /Frame )
   rlay = WwLayout( layout, /Top, /Bottom, /Form, /Frame, Left=llay )
```

```
    dlay = WwLayout( layout, /Top, /Right, /Bottom, left=rlay, /Form,
    /Frame )

     grael_name = TmSetAttribute( tool_name, 'TM', 'CURR_GRAEL', $
         'GET_PARM_STATS_USR' )
     ifc = TmSetAttribute( tool_name, 'GET_PARM_STATS_USR', $
         'CURR_IFC', 'B' )
     curr_axis = TmSetAttribute( tool_name, 'GET_PARM_STATS_USR', $
         'CURR_AXIS', 'GET_PARM_STATS_USR_1' )

     parm_wids = GU_newParmlist_component( tool_name, parent=llay )
     rv = GU_updateParmlist_component( tool_name, wids=parm_wids )

     ; text area
     wid = WwText( rlay, 'NoOpCB', Cols=30, Row=35, $
                   Foreground='Black', Background='', /Bottom )

     ; drawing area
     windowID = -1
     wdraw = WwDrawing( dlay, windowID, 'udemo_TSDrawCB', $
                        [512,512], [512,512], $
                        area=area, /noscroll )

    whandler = WwHandler( area,'udemo_handlerCB', 'ButtonPressMask' )

     rv = TmSetAttribute( tool_name, 'PARM_STATS_WIDS', 'TEXT_WID',
    wid )

     ; Save widget ids to set current values when OK or Apply selected
     FOR i = 0, 1  DO $
         rv = WwSetValue( buttons(i), User-
    data=LIST(tool_name,parm_wids) )

     status = WwSetValue( dialog, /Show )
END
```

Tips for Creating User Functions

User functions can be as simple or as complex as you choose. The TS-WAVE utilities listed below are useful in developing user functions. For more information, see *Writing a User Function* on page 40.

- **Dm_getSrcList** — Returns a list of all opened Source names or data files.
- **Dm_getParmList** — Returns a list of available parameter names for a specified source name.
- **Dm_getParm** — Returns the data values and time in an Associative array. Dm_getParm is needed to create derived parameters.

In addition, TS-WAVE uses three routines that can be called to create the following: same Source selection, a parameter check, and a parameter selection interface component.

- **Gu_newParmlist_Component** — Creates graphical interface component.
- **Gu__updateParmlist_Component** — Updates with latest source files.
- **Gu__readParmlist_Component** — Reads interface to get the selected source name and parameter name.

TS-WAVE DataHandlers

General

TS-WAVE DataHandler functionality provides a robust mechanism for you to integrate your data file formats with TS-WAVE. The DataHandler concept consists of an abstract interface that allows you to easily make your data accessible with TS-WAVE. Defined methods provide the framework that makes it easier to implement input and output of your data format. These methods are developed outside of TS-WAVE allowing for a graphical user interface tailored specifically for the user selection of pertinent information within your data file. See *Creating a Datahandler* on page 60.

TS-WAVE relies on the *datahandler.ads* string file to obtain and modify DataHandler information, such as the default directory location, and a list of DataHandlers. The list of DataHandlers is especially significant in that it contains ‘all’ available DataHandlers for the TS-WAVE session, and their directory locations.

NOTE All resource files are located in **TS-WAVE_DIR\resource.**

Example

This example was extracted from the *datahandler.ads* resource file.

```
!-------------------------------------------------------------
!  datahandler.ads  String Resource File
!-------------------------------------------------------------
DH_Root:
DH_Subdir:        datahandlers
DH_Version: 3.0
DH_TypeListFile: datahandler.list
DH_TypeDir_Separator: :
```

In the example above, `DH_Root` is not set. `DH_Root` is set as the default, `TS-WAVE_DIR`. `DH_Subdir` is set to DataHandlers. `DH_TypeListFile` is set to the default filename *DataHandler.list*. When TS-WAVE starts it searches in `TS-WAVE_DIR/datahandlers/datahandler.list` for all DataHandlers and registers them with the application. `DH_TypeDir_Separator` prompts TS-WAVE to use the symbol '**:**' as the separator between the DataHandler name and the directory in which the DataHandler source code is found.

`DH_Root` assigns the top level Datahandler directory, by default this is set to `VNI_DIR/tswave-3_0` as in the example above. `DH_Subdir` assigns the subdirectory location where all common DataHandlers directories and files are located. By default this is `VNI_DIR/tswave-3_0/datahandlers` as above. `DH_TypeListFile` sets the name of the file containing all available DataHandlers and their directory locations. `DH_TypeDir_Separator` allows the user to specify a different separator symbol to delineate between the DataHandler name and the directory. In the example above, the settings prompt TS-WAVE to look for `VNI_DIR/tswave-3_0/datahandlers/datahandler.list` to gather the list of DataHandlers for the TS-WAVE session.

Example

Here is an example of the *DataHandler.list.*

```
ldf   : $TSWAVE_ROOT/datahandlers/ldf
ascii : $TSWAVE_ROOT/datahandlers/ascii
```

```
tab   : $TSWAVE_ROOT/datahandlers/tab
```

Where,

ldf is the name of a specific DataHandlers; and the symbol '**:**' is the separator. `c:/ts_wave-3_0/datahandlers/ldf` is the directory location of the *ldf* DataHandler methods.

After TS**-**WAVE has read the list of available DataHandlers it registers them with the application. If you choose **File=>Open** from the TS**-**WAVE menu you are prompted to choose from a list of format types which correspond to the list of DataHandlers in the *DataHandler.list.* The format specified will determine which DataHandler is called to open the data file selected by the user.

NOTE PV**-**WAVE functions `<dhtype>_readfile.pro` and `<dhtype>_readdata.pro` must be located in each DataHandler directory for TS**-**WAVE to work.

`<dhtype>_readfile.pro` opens the file and initializes any settings needed to read data. `<dhtype>_readfile` can be used to display a user interface that allows users to select settings specific to the format, such as select a time slice from the data file, a subset of parameters, or sampling rates. The primary function of `<dhtype>_readfile` is to set the *Parameter* list, which allows a user to choose a parameter from the file. Other useful settings are start and stop time, and units. `<dhtype>_readdata` must provide TS**-**WAVE with data values for a specific parameter. A third routine, `<dhtype>_writefile.pro`, allows data to be written to a specific format.

NOTE *<dhtype>* refers to the name of the DataHandler as listed in *datahandler.list*.

DataHandler Requirements

TS**-**WAVE DataHandlers rely on several PV**-**WAVE functions to work. If any of these requirements are not met, the DataHandlers will not function from within TS**-**WAVE.

A DataHandler must have the following:

- ***<dhtype>_READFILE.PRO*** — Opens the a data file of *<dhtype>* format. Returns the names of all available parameter names.

- ***<dhtype>_READDATA.PRO*** — Reads the data file of <*dhtype*> format and returns the data values and time values for a given parameter name.
- ***<dhtype>_WRITEFILE.PRO*** — *[optional]* Writes out data to a specific <*dhtype*> file format. For example, the *ldf* file format requires *ldf_ReadFile.pro*, *ldf_ReadData.pro*, and *ldf_WriteFile.pro*

<dhtype>_ReadFile.pro

Usage

status = <*dhtype*>_READFILE(*fileinfo, tool_name*)

Input Parameters

fileinfo — A PV-WAVE associative array containing information about the source file opened. The information is set by the calling function.

tool_name — A string containing the unique name of a VDA Tool. The information is set by the calling function.

Returned Value

status – An integer containing a 1 if successful and less than 1 if unsuccessful.

Discussion

This routine reads and gathers any header and any setup information that is needed to read parameter data from the file at the time in which the parameter was requested.

dhtype_ReadData.pro

Usage

data = <*dhtype*>_READDATA(*fileinfo, parm, tool_name*)

Input Parameters

fileinfo — A PV-WAVE associative array containing information about the source file opened.

parm — String containing the parameter name.

tool_name — String containing the unique VDA Tool name.

Returned Value

data — A PV-WAVE associative array with one required KEY: ***data*** — an array of required data values; and one optional KEY: ***time*** — an optional PV-WAVE DT array of times corresponding to the data values.

Discussion

This routine retrieves parameter data values and optional time values associated with the data.

dhtype_WriteFile.pro

Usage

status = <*dhtype*>_WRITEFILE(*tool_name*, *filename*)

Input Parameters

tool_name— String containing the unique VDA Tool name.

Filename— String containing the filename selected from the file selection interface.

Returned Value

status — An integer containing a 1 if successful and less than 1 if unsuccessful.

Discussion

TS-WAVE's unique tool name and the file name selected from the File Selection interface is passed as the input parameter for this routine to be used to generate a file of <*dhtype*>'s format.

FileInfo Associative Array

The section describes the standard keys in the *FileInfo* associative array.

CAUTION The standard keys are case sensitive.

parameter_list — An array of strings containing a list of available parameter names.

NOTE ***parameter_list*** must be set

units — A string array of units corresponding to the array of parameters.

dims — String array of units corresponding to the array of parameters.

file_fullname — String containing the directory and name of the source file. TS-WAVE handles user selecting filename.

filepath — Path of the file selected.

filename — Filename of the file selected.

file_size — Optional setting.

file_sample_rate — An array of sample rates corresponding to each parameter.

start_time — PV-WAVE DT structure containing the start time of the file.

stop_time — PV-WAVE DT structure containing the stop time of the file.

dt_base — DT structure used to convert time. If this is not set PV-WAVE's default !Dt_base value is used for all time conversions.

Creating a Datahandler

TS-WAVE's DataHandlers are a group of PV-WAVE functions that collect data for a specific file format. Each DataHandler must have a unique name, indicated here

by <*dhtype*>. For example, the unique name for the DataHandler that is used to read the *Loral Data Format* is called *ldf*.

NOTE Functions that implement DataHandler functions begin with <*dhtype*>.

As mentioned earlier, TS-WAVE uses the following DataHandler functions:

- <*dhtype*>_READFILE
- <*dhtype*>_READDATA
- <*dhtype*>_WRITEFILE

The steps below describe how to create a datahandler to read a file format.

1. Decide on a *name* for your DataHandler. For example, *ex1*.

NOTE We will use DataHandler *ex1* throughout the remaining steps and examples.

---

2. Create a function called *ex1*_readfile.pro.

```
; Setup Info
FUNCTION ex1_readfile, fileinfo, tool_name
Return, status;1: success  0: failure
END
```

3. Next, access the file name assigned by user through a file selection widget invoked during a TS-WAVE session using **File=>Open**.

```
FUNCTION ex1_readfile, fileinfo, tool_name
Filename = fileinfo('FILE_FULLNAME')
;Open file
...
;Read file
...
Return, status;1: success  0: failure
END
```

4. Also within *ex1*_Readfile assign a parameter name(s) to `fileinfo` to be used by TS-WAVE.

```
FUNCTION ex1_readfile, fileinfo, tool_name
```

```
Filename = fileinfo('FILE_FULLNAME')
;Open file
...
;Read file
...
;Assign parameter names to variable parmlist
...
fileinfo('PARAMETER_LIST') = parmlist

Return, status;1: success  0: failure
END
```

5. Likewise, assign start and stop time.

```
fileinfo('START_TIME') = starttime
fileinfo('STOP_TIME') = stoptime
```

6. Assign any other optional settings in the `fileinfo` structure that may pertain to information available within the datafile.

```
fileinfo('UNITS') = units
fileinfo('DESAMPLE') =desample_rate
fileinfo('DT_BASE') = dt_base
fileinfo('DIMS') = parm_dimensions
```

7. Now, let's assign DataHandler-specific information in `fileinfo`. Since `fileinfo` is an associative array, your DataHandler can add as many additional keys as needed. This information is sent to the *ex1*_ReadData function. Also, any information that is needed to acquire parameter data can be added to `fileinfo`.

```
fileinfo('CURR_RUN') = selected_run
fileinfo('offset') = fileoffset
```

8. Return status of completion. 1 if successful, and 0 if unsuccessful.

```
FUNCTION ex1_readfile, fileinfo, tool_name
Filename = fileinfo('FILE_FULLNAME')
;Open file
...
;Read file
...
;Assign parameter names to variable parmlist
```

```
...
fileinfo('PARAMETER_LIST') = parmlist
...
Return, status;1: success  0: failure
END
```

9. Now we need to create a *ex1*_readdata.pro function.

```
FUNCTION ex1_readdata, fileinfo, param, tool_name
 Return, ASARR('DATA',values,'TIME',time)
END
```

10. Use the information in `fileinfo` to gather data for the parameter specified in the second positional parameter. Return an associative array with the data values and time array for the parameter. If time values do not exist for the parameter, return an indexed array of integers with the same number of elements as the parameter's data values.

```
Return, ASARR('DATA',values,'TIME',indgen(n_elements(values)))
```

NOTE Keywords 'DATA' and 'TIME' must be CAPITALIZED.

```
FUNCTION ex1_readdata, fileinfo, param, tool_name
Filename=fileinfo('FILE_FULLNAME')
Offset=fileinfo('offset')
...
; Read data and time in variables values and time
...
; convert time to a DT struct
...
 Return, ASARR('DATA',values,'TIME',time)
END
```

11. Add *ex1* and the path to the *ex1* directory (where `ex1_readfile.pro` and `ex1_readdata.pro` are located) to the list in the file *datahandlers.list* . By default, this list is located in the `$TSWAVE_ROOT/datahandlers` but may be elsewhere if defined in `$TSWAVE_ROOT/resources/datahandlers.ads`. TS-WAVE automatically adds *ex1*'s directory to PV-WAVE's path.

```
ldf   : $TSWAVE_ROOT/datahandlers/ldf
ascii : $TSWAVE_ROOT/datahandlers/ascii
```

```
tab   : $TSWAVE_ROOT/datahandlers/tab
ex1 : $TSWAVE_ROOT/datahandlers/example
```

TS-WAVE Utility Routines

General

TS-WAVE's DataManager and User Interface utility routines make it easy to manage data and integrate elements into a graphical user interface (GUI).

The DataManager routines (DM_*) make it easier to handle data and build interfaces for your user function. The DataManager routines (DM_*) include functions that allow a derived parameter to be accessed easily from other TS-WAVE routines, give you access to the source list and parameter list, and let you retrieve data values associated with a parameter (including properly created derived parameters). The User Interface routines (GU_*) allow easy integration of elements into a GUI, and can also create a list widget and later update or retrieve parameters contained in it.

TS-WAVE's DataManager Routines

TS-WAVE's DataManager routines are:

- *DM_addFCN Function*
- *DM_getSrcList Function*
- *Dm_getSrc Function*
- *DM_getParmList Function*
- *DM_getParm Function*

DM_addFCN Function

Adds the derived parameter to the Function List, making the derived parameter available to other TS-WAVE functions.

Usage

rv = DM_addFCN(*tool_name*, *newparm*, *exe_cmd*)

Input Parameters

tool_name — String containing the unique VDA tool name.

newparm — Name of the new derived parameter.

exe_cmd — String containing the commands to create the derived parameter. See PV-WAVE's EXECUTE command.

Keywords

None

Returned Value

rv — Integer value indicating success or failure.

0 — Operation failed to add derived parameter to Function List.

1 — Operation succeeded in adding derived parameter to Function List.

Example

Here's part of an example taken from the complete user function example that appears on page 44.

```
ret_data = GU_readParmlist_component( tool_name, wids=parm_wids )
parm_name = ret_data('PARAMETER')
src_name  = ret_data('SOURCE')
newparm = 'UDemo_Usr(' + STRTRIM(src_name(0),2)+':'+parm_name(0) +
   ')'
exe_cmd = "UDemo(tool_name,'" + src_name + "','" + parm_name + "')"
rv = DM_addFCN(tool_name, newparm, exe_cmd)
```

DM_getSrcList Function

Returns a string array of available data source identification names for the TS-WAVE session.

Usage

src_list = DM_getSrcList(tool_name)

Input Parameters

tool_name — String containing the unique VDA tool name.

Keywords

None

Returned Value

src_list — String array containing the list of open source file names.

Dm_getSrc Function

Returns an associative array containing all of the available information for the data source.

Usage

fileinfo = DM_getSrc(*tool_name, src_name*)

Input Parameters

tool_name — String containing the unique VDA Tool name.

src_name — String containing the name of the source file.

Keywords

None

Returned Value

fileinfo- An associative array containing file information, such as 'FILE_FULLNAME' which contains the file location, 'FILETYPE' containing the format type, and 'PARAMETER_LIST' containing the list of available parameters. See FileInfo Associative Array *on page 60* for a list of associative array keys. If the routine is unsuccessful an integer of value less than 1 will be returned.

DM_getParmList Function

Returns a string array of available parameters for a given source.

Usage

parm_list = DM_getParmList(*tool_name*, *src_name*)

Input Parameters

tool_name **—** String containing the unique VDA tool name.

src_name **—** String containing the name of the source file.

Keywords

None

Returned Value

parm_list — String array containing the names of the available parameters.

DM_getParm Function

Returns the data and time values associated with a specified parameter from a specified data source.

Usage

data_values = DM_getParm(*tool_name*, *src_name*, *parameter*)

Input Parameters

tool_name — String containing the unique VDA tool name.

src_name — String containing the name of the source file.

parameter — String containing the parameter name.

Returned Value

data_values — An associative array with the following required KEYS:
data — an array of scalar data values;
time — a PV-WAVE DT (date-time) array of times corresponding to the data values. Or array of scalar values with same number of elements as data values.

TS-WAVE's User Interface Routines

TS-WAVE's User Interface routines are:

- *GU_newParmlist_Component Function*
- *GU_updateParmList_Component Function*
- *GU_readParmlist_Component Function*

GU_newParmlist_Component Function

Creates a new parameter list in an existing layout.

Usage

wids = GU_newParmlist_Component(*tool_name*, parent=parent)

Input Parameters

tool_name — String containing the unique VDA tool name.

Keywords

parent — The widget ID of the parent widget, in most cases, a layout ID.

Returned Value

A structure containing the widget IDs created. If the function fails, a value less than 1 is returned. These widget IDs can be used to directly interact with the elements of the parameter list component. The table below lists the tags and their associated objects.

Tag Name	Object
parmtxt_id	Text ID for selected parameter
srclist	Source Selection List ID
parmlist_id	Parameter List ID
sort_button_id	Sort parmlist button ID

NOTE Once the main component has been created, GU_updateParmlist_Component must be called to populate the widgets with source names and parameters.

GU_updateParmList_Component Function

Updates the Parmlist_Component with current source files and parameters. If source is not supplied, the first source file name in the list is selected and its parameters are used to populate the *Parameter* list box.

Usage

rv = GU_updateParmlist_Component(*tool_name*, *wids=wids*, *src=src*, *parm=parm*)

Input Parameters

tool_name — String containing the unique VDA tool name.

Keywords

wids — Structure of widgets returned by the GU_newParmlist_Component routine.

src — Optional keyword to pass a specified source name to update the *Parameter* list widget.

Parm – Optional keyword to pass a specified paramater to be highlighted in the *Parameter* list widget.

Returned Value

rv — Integer value equal to 1 for success, if the function fails, a value less than 1 is returned.

GU_readParmlist_Component Function

Reads the *Parmlist_Component* to obtain the selected Source name and Parameter name.

Usage

rv = GU_readParmlist_Component(tool_name, wids = *wids*)

Input Parameters

tool_name — String containing the unique VDA tool name.

Returned Value

rv — Associative array containing the source and parameter names selected in the ParmList_Component in the keys SOURCE and PARAMETER.

Keywords

wids — Structure of widgets returned by the GU_newParmlist_Component routine.

Addtional GU_readParmlist_Component Information

When retrieving from a calling procedure:

Source_name = ***rv***(‘SOURCE’)

Parm_name = ***rv***(‘PARAMETER’)

3

CHAPTER

GUI Reference

This reference discusses:

- *TS-WAVE Menus*
- *Dialog Boxes*
- *Miscellaneous Topics*

TS-WAVE Menus

This section describes the commands on the TS-WAVE menus. The menus include:

File Menu

New

Starts a new TS-WAVE session. Existing graphics are deleted. Current data remains loaded.

CAUTION Be sure to save existing graphics before choosing the **New** function. Otherwise, your current, unsaved work will be lost.

Page Setup

Brings up the Page Setup dialog box. The Page Setup commands change the orientation and size of the TS-WAVE drawing area. These commands affect both the on-screen and printed versions of the drawing area.

For information on the functions in the Page Setup dialog, see *Page Setup Dialog Box* on page 106.

Open

Brings up the Open Data File dialog box, which is a standard file selection tool. Use this dialog to select a file to open. When you click **OK** in this file selection tool, the Name and Type Data Source dialog box appears. Use this dialog to specify the data run to display, desampling rate, samples per second, and other parameters.

Close

Brings up the Close Source Files dialog box. Use this dialog to close any or all source files in the current session.

Export

Brings up the Export Data dialog box, which is a standard file selection tool. Use this dialog to select a filename in which to export data. When you click **OK** in the file selection tool, the Select Export Format dialog box appears. Choose a format type of your choosing to export your data. The next step varies for each format type. Not all formats will have export ability.

Save Template

Brings up the Save Template As dialog box. This dialog is a standard file saving tool that allows you to choose the filename and directory in which to save the template file.

A template is a file that allows you to reuse the configuration of previously created TS-WAVE plots. Everything in the session *except the data* is saved in the template file. For instance, if you create a graph that contains characteristics that you wish to reuse, such as parameter names, headers, colors, fonts, graph sizes, and graph positions, you can save this configuration in a template. Then, whenever you open

the template, the same plot objects appear in the same configuration as they were when the template was saved. You can then import data into the object(s) of your template simply by loading the data into TS-WAVE.

To open a previously saved template, use the **File=>Open Template** command.

By default, template files are given a `.tpl` extension (for example, `myfile.tpl`).

NOTE A template differs from a session file. A session file contains the same information as a template described previously. Also included is the data that was plotted when the session was saved.

Save Session

Brings up the Save Session As dialog box. This dialog is a standard file saving tool that allows you to choose the filename and directory in which to save the file.

A session file contains the complete TS-WAVE plot in exactly the state it was in when it was saved, including any data that was plotted. To open a saved session file, use the **File=>Open Session** command.

NOTE A session save file differs from a template file. A template file saves only the configuration of the session, not the actual data.

Open Template

Brings up the Restore Template dialog box. Use this standard file opening tool to open a previously saved template file. For more information on template files, see the **Save Template** command.

Open Session

Brings up the Restore Session dialog box. Use this standard file opening tool to open a previously saved session file. For more information on session files, see the **Save Session** command.

Print

Prints the contents of the drawing area (the current plots). If you are printing for the first time, the Printer Selection dialog box appears. Use this dialog to specify the name and type of printer. For more information on this topic, see page 107.

Print Setup

Brings up the Printer Setup dialog box. This dialog lets you specify the name and type of printer. You can also use this dialog to print to a file. For information on this dialog, see *Printer Setup Dialog Box* on page 106.

Exit

Exits TS-WAVE.

CAUTION Be sure to save your work before you select the **Exit** command. Otherwise, your current, unsaved work will be lost.

NOTE (*UNIX only*) You must also type `EXIT` at the `WAVE>` prompt in the TS-WAVE command window to exit the PV-WAVE session.

Edit Menu

Object Select

Lets you select graphical objects in the plot using the mouse. A graphical object includes anything that was created by commands on the **Create** menu: graphs, contours, headers, text, lines, boxes, and circles. Selected graphics objects can be cut, copied, pasted, deleted, or grouped. Furthermore, if you wish to modify the attributes of a graphics object, you must select it first.

For more information, see *Selecting, Resizing, and Removing a Graph* on page 10.

Cut

Cuts the selected graphical object(s) to the clipboard.

NOTE The clipboard is an unseen location for temporary storage of the graphics object(s) from the last **Cut** or **Copy** operation. The contents of the clipboard can be "pasted" back to the graphics area. Each new **Cut** or **Copy** overwrites the current contents of the clipboard.

Copy

Copies the selected graphical object(s) to the clipboard.

Paste

Pastes the contents of the clipboard in the plot window. The pasted object is placed in front of other objects in the drawing area.

Delete

Removes the selected graphical object(s) from the plot window and redraws the view. A deleted object cannot be retrieved.

Align Graphs

Use this function to align two or more graph objects. First, select one graph object as the "master" graph, then select **Align Graphs**. All other graph objects are aligned relative to the master graph. In addition, the *x*-axes of all graph objects are scaled to the size and axis range of the *x*-axis of the master graph object.

For more information on this function, see *Moving and Aligning Graphs* on page 25.

Set X-Axis Range

Sets the x-axis range in your session to match the axis you selected in the interface.

Plot Attributes

Brings up the Graph Attributes Interface dialog box. Use this dialog to plot attributes for the selected graph. See *Graph Attributes Interface Dialog Box* on page 92.

Select All

Selects all graphical objects in the drawing area.

Deselect All

Deselects all graphical objects in the drawing area.

Redraw

Redraws the drawing area.

Group

Groups the selected graphical objects. Grouped items can be cut, copied, pasted, deleted, and moved as one unit.

TIP To select multiple objects: hold down the <Shift> key and click on the objects you wish to select; or, choose **Edit=>Select All** to select all objects in the drawing area at once; or, press and hold the left mouse button and drag the pointer around the objects you wish to select.

Ungroup

Ungroups grouped graphical objects.

Front

Brings the selected graphics object(s) to the front of the drawing area.

Back

Sends the selected graphics object(s) to the back of the drawing area.

View Menu

Page Zoom

This command is a pixmap zoom that controls the magnification of the drawing area. Choose **View=>Page Zoom** to enter a desired zoom factor from 0.1 to 3.0 in the dialog box that appears. For more information, see page 108.

Actual Size

Sets magnification so the actual page size is visible in the drawing area.

TIP The Page Zoom feature does not allow for changes to the displayed axis range. Use the Data Zoom feature if you desire to change and apply a newly zoomed range to existing(s) graph objects.

TIP If the drawing area does not refresh correctly, click in one or both of the scroll bars, or select **Edit=>Redraw**.

Data Zoom

This command allows TS-WAVE users to interactively zoom on one or more parameters, effectively setting the X and Y ranges for the parameters, using the mouse. For more discussion on this topic, see page 20.

- For step-by-step instructions, see the tutorial section on page 20.
- For an in depth discussion on all the Data Zoom features, see page 108.

Show Auto Scale

Auto scaling looks at a range and calculates the data for you based on a parameter of your choosing.

Show Grid Selections

These menu selections add grid squares to the graphs in the drawing area. You can add or remove the grids at any time. All graphs in the drawing area are immediately updated with the new grid dimension(s) after you make your choice. The choices are:

Show CM Grid — One centimeter squares. (Default)

Show Half-CM Grid — One-half centimeter squares.

Show MM Grid — One millimeter squares.

Show No Grid — Do not display a grid in the graph objects.

Print Grid Selections

This command is similar to the **Show Grid** command; however, **Print Grid** only applies to printed graphs (the screen display is not affected). You can change the print grid dimensions at any time. The choices are:

Print CM Grid — One centimeter squares. (Default)

Print Half-CM Grid — One-half centimeter squares.

Print MM Grid — One millimeter squares.

Print No Grid — Do not display a grid in the printed graph objects.

CM Grid Color

Brings up the Select CM Grid Color dialog box. Use this dialog to change the color of the CM (one centimeter square) grids. This command changes the color of all

CM grids currently displayed in the drawing area, the print color, and the default grid color.

HCM Grid Color

Brings up the Select HCM Grid Color dialog box. Use this dialog to change the color of the HCM (half centimeter square) grids. This command changes the color of all HCM grids currently displayed in the drawing area, the print color, and the default grid color.

MM Grid Color

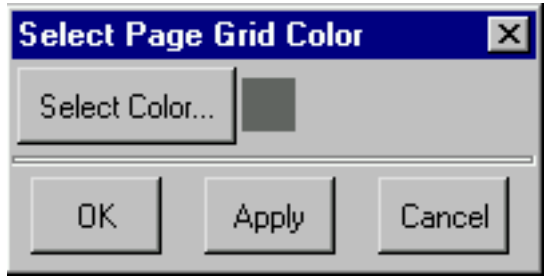
Brings up the Select MM Grid Color dialog box. Use this dialog to change the color of the MM (one millimeter square) grids. This command changes the color of all MM grids currently displayed in the drawing area, the print color, and the default grid color.

Page Grid

Displays a grid on the entire drawing area.

Page Grid Color

Brings up the Select Page Grid Color dialog box. Use this dialog to change the color of the page grid.



The Select Page Grid Color dialog box

Background Color

Brings up the Select Background Color dialog box. Use this dialog to change the background color of the drawing area. This command changes the background color in the current drawing area, the print color, and the default background color.

Foreground Color

Brings up the Select Page Grid Color dialog box. Use this dialog to change the foreground color of the drawing area (for displayed and printed versions). Foreground objects include axes, axis titles, and the information block at the bottom of the drawing area. This command changes the foreground color of all graphs currently displayed in the drawing area, the print color, and the default foreground color.

TIP Make sure the background and foreground colors are not the same.

Show Info

Select this option to display information about the plotted data run in the lower-left corner of the drawing area (and on printed copy). The information includes the start and stop time of the run, the run number, the path and filename of the data file, and the template name.

Auto Redraw

If this option is selected, the drawing area is redrawn whenever a change is made. This feature is disabled when a batch file is opened for creation.

Auto Load

If this option is selected, any new data that is selected is automatically displayed on the current graph. Otherwise, the graph is not updated when new data is selected. This feature is disabled when a batch file is opened for creation.

Create Menu

Graph Object

Lets you add a new graph object to the drawing area. The general procedure for adding a graph object is described in *Creating a Graph* on page 9.

Header Object

Lets you add a header object to the drawing area. A header object is simply a box that contains text pertaining to a graph. The general procedure for adding a header object is described in *Adding a Header* on page 12.

Contour Object

Lets you add a contour object to the drawing area. The general procedure for adding a contour object is described in *Creating a Contour* on page 18. For an in-depth discussion, see *Contour Attributes Interface Dialog Box* on page 98.

Text Object

Lets you add words or phrases to accentuate important trends and differentiate features of a plot. The general procedure for adding text to the drawing area includes these steps:

1. Select **Create=>Text Object**.

2. Click where you want the text to start.

3. Enter the text and, optionally, specify text characteristics (font, color, and so on) in the Text Attributes dialog box.

4. Click **OK** when you are finished entering the text. You can continue to create text objects until you make another selection from the **Create** menu.

Line Object

Lets you add lines to emphasize a particular value or to visually connect text to some feature of interest elsewhere in the plot. The general procedure for adding lines to the drawing area includes these steps:

1. Select **Create=>Line Object**.

2. Use the mouse to draw the line. Do this by pressing the left mouse button and dragging the pointer. When you release the mouse button, the line is fixed and you can draw another line if you wish. You can continue to draw lines until you make another selection from the **Create** menu.

Box Object

Lets you add a box to delineate a region of special interest. The general procedure for adding boxes to the drawing area includes these steps:

1. Select **Create=>Box Object**.

2. Use the mouse to draw the box. Do this by pressing the left mouse button and dragging the pointer. When you release the mouse button, the box is fixed and you can draw another box if you wish. You can continue to draw boxes until you make another selection from the **Create** menu.

Circle Object

Lets you add a circle to delineate a region of special interest. The general procedure for adding circles to the drawing area includes these steps:

1. Select **Create=>Circle Object**.

2. Use the mouse to draw the circle. Do this by pressing the left mouse button and dragging the pointer. As you drag the pointer, a box-shape indicates the position and shape that the circle circumscribes. When you release the mouse button, the circle is fixed, and you can draw another circle if you wish. You can continue to draw circles until you make another selection from the **Create** menu.

Create TabData

This command lets you save parameters in a tabular data file, which can be used for reports or read into TS-WAVE.

- The procedure for creating a tabular data file is described in *Creating a Tabular Data File* on page 26.
- See also *Select Tab Attributes Dialog Box* on page 89 and *Select Tab Attributes Advanced Settings Dialog Box* on page 91.

View TabData File

Brings up a standard file opening dialog box. Use this dialog to select the tabular data file that you wish to open. The file is displayed in a text window. For more information, see *Creating a Tabular Data File* on page 26.

NOTE The default View TabData text window uses proportional fonts, which can cause some irregularity in the alignment of data in the columns. When you print your data file, the columns will be aligned properly. To see properly aligned columns on your screen, you must use a text editor that is set to use a non-proportionally spaced font. To change the default viewer, see the README file under the TS-WAVE resource directory.

Open Pick File

Use this command to place data points at selected time slices in a tabular data file. For detailed information on this command, see *Creating a Pick File* on page 32.

Close Pick File

Closes the pick file, and the vertical line used for picking data disappears from the drawing area.

Open Batch File

Brings up a standard file naming tool. Specify a name and directory for the batch file and click **OK**.

Add Template to Batch

Loads a template and adds it to the batch file. For general information on templates, see *Save Template* on page 72.

Close Batch File

Saves the batch file and exits the batch file creation mode.

Analyze Menu

NOTE For more information on using the standard functions, see *Using Standard Functions* on page 22.

Standard=>Bias

Brings up a dialog that lets you add a bias value to a parameter. The result of the operation is a new parameter; the original parameter is unchanged. The new parameter is named:

Bias(*src*1:ALTB, 5.00000)

where *src*1 is the name of the source id, and *ALTB* is the parameter selected from the ‘Select Parameter to Bias’ interface dialog box, and 5.00000 is the bias value. After the bias value is added the parameter is placed into the ‘DERIVED’ data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Standard=>Difference

Brings up a dialog box that lets you compute the difference between two selected parameters. The difference is computed by subtracting the value of the selected parameter in the right column from the value of the selected parameter in the left column.

The result of the operation is a new parameter; the original parameter is unchanged. The new parameter is named:

DIFFERENCE(*src*1*:*ASPLEU, *src2:*CTXSBP)

where *src*1 and *src*2 are the names of the source ids, and ASPLEU and CTXPB are the parameters selected from the 'Select Different Parameters' dialog box. CTXSBP values are subtracted from ASPLEU values. After the difference is computed the parameter is placed into the 'DERIVED' data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Standard=>Differentiate

Brings up a dialog box that lets you compute the derivative of a specified parameter.

The result of the operation is a new parameter; the original parameter is unchanged. The new parameter is named:

DIFFERENTIATE(*src*1*:*CTXSPS)

where *src*1 is the name of the source id, and CTXSPS is the parameter selected from the 'Select Parameter to Differentiate' dialog box. The derivative of CTXSPS is computed. After the derivative is computed tthe parameter is placed into the 'DERIVED' data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Standard=>FFT

Brings up the dialog that lets you return the fast Fourier transform (FFT) for the input variable. For detailed information on the FFT function, see *PV-WAVE User's Guide*. The new parameter is named:

FFT*(src*1*:*ASPREU)

where *src*1 is the name of the source id, and ASPREU is the parameter selected from the ‘Select Parameter to FFT’ dialog box. The FFT of ASPREU is computed. After the FFT is computed the parameter is placed into the ‘DERIVED’ data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Standard=>Gain

Brings up a dialog that lets you apply a gain value to a selected parameter. The result of the operation is a new parameter; the original parameter is unchanged. The new parameter is named:

GAIN(*src*1:ESPLEU, 2.00000)

where *src*1 is the name of the source id, and ESPLEU is the name of the parameter selected from the ‘Select Parameter to Apply Gain to’ dialog box. The ESPLEU values are multiplied by the gain value 2.00000. After the gain value is applied the parameter is placed into the “DERIVED” data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Standard=>Gauss Fit

Brings up a dialog that lets you fit a Gaussian curve through the data points of the original parameter. The result of the operation is a new parameter; the original parameter is unchanged. The new parameter is named:

GAUSSFIT(*src*1:COLFEU)

where *src*1 is the name of the source id, and COLFEU is the name of the parameter selected from the ‘Select Parameter to GAUSSFIT’ dialog box. The Gaussian curve of COLFEU is computed. After the Gaussian curve is computed the parameter is placed into the “DERIVED” data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Standard=>Smooth

Brings up a dialog that lets you smooth the data points in the original parameter. The result of the operation is a new parameter; the original parameter is unchanged. The new parameter is named:

SMOOTH(*src*1:DPLOEU,5.00000)

where *src*1 is the name of the source id, and DPLOEU is the name of the parameter selected from the 'Select Parameter to Smooth' dialog box. The values in DPLOEU are smoothed with a boxcar average of 5.00000. After the data points are smoothed the parameter is placed into the 'DERIVED' data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Standard=>Trim

Brings up a dialog that lets you create a new parameter whereby the initial value of the original parameter is subtracted from each subsequent point in the original parameter. The result of the operation is a new parameter; the original parameter is unchanged. The new parameter is named:

TRIM(*src*1:ALTS)

where *src*1 is the name of the source id, and ALTS is the name of the parameter selected from the 'Select Parameter to Trim' dialog box. The values in ALTS are subtracted from the initial value in ALTS. After subtraction the parameter is placed into the 'DERIVED' data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Standard=>WildPoint

Brings up a dialog that lets you create a new parameter by removing data points that deviate from specified limits in the original parameter. The result of the operation is a new parameter; the original parameter is unchanged. The new parameter is named:

WildPoint(*src*1:BCAS,10.000000,20.000000)

where *src*1 is the name of the source id, and BCAS is the name of the parameter selected from the 'Select Parameter to WildPoint' dialog box. The values between

10.000000 and 20.000000 are extracted from BCAS. After the extracted vlaues are removed the parameter is placed into the ‘DERIVED’ data source parameter list.

To access the new parameter, select a graph object, and then double click on the graph to bring up the Graph Attributes Interface dialog box. Select **DERIVED** from the Data Source menu. The new parameter appears in the list of parameters.

Analyze=>User

Use this menu to access user-defined TS-WAVE functions. For information on creating functions and adding them to this menu, see Adding User-defined Functions on page 27.

Help Menu

On-line Help

Brings up the on-line version of the TS-WAVE User’s Guide.

Version

Displays information on the current version of TS-WAVE.

User Fcn Help

Displays help on user-created functions. For information on adding functions to this menu, see *Using User Functions* on page 24.

About TS-WAVE

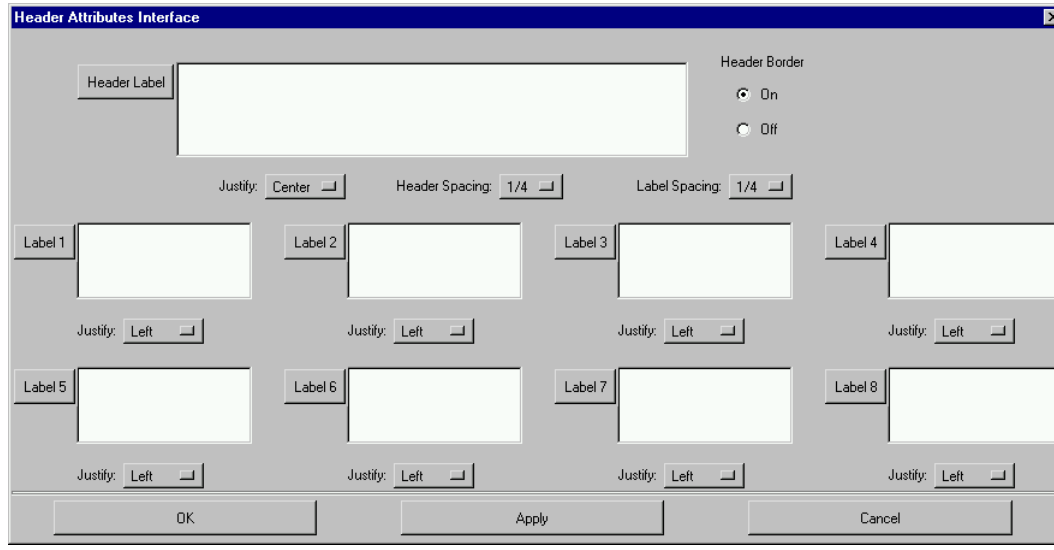
Displays information about TS-WAVE.

Dialog Boxes

This section describes the contents of the following dialog boxes:

Header Attributes Interface Dialog Box

Use this dialog to specify labels for a header. This dialog also allows you to include calculations in a header using PV-WAVE functions.



The Header Attributes Interface dialog box

Header Label — Use this field to enter the text you wish to appear in the main header, or title, on your drawing area. Click the **Header Label** button to bring up a dialog that lets you modify the appearance of the text (font, text size, text position, and so on). You can embed PV-WAVE functions in labels—see the note at the end of this section.

Header Border — Lets you specify whether or not the header has a border around it.

Justify — Lets you specify **Left**, **Center**, or **Right** justification for the header label within the header box.

Header Spacing — Lets you specify the amount of space (in centimeters) you wish to surround the header label.

Label Spacing — Lets you specify the amount of space (in centimeters) you wish to surround all other labels.

Label 1–8 — Use this field to enter additional labels in these text fields. Click the Label # button to bring up a dialog that lets you modify the appearance of the text (font, text size, text position, and so on).

OK — Exits the dialog box and applies the settings.

Apply — Applies the settings, but does not exit the dialog box.

Cancel — Exits the dialog without making any changes.

NOTE You can embed PV-WAVE functions in TS-WAVE labels. The functions can perform any valid operation on any TS-WAVE parameter.

For example, you could embed a function in the header that computes the average reading in the ALTB parameter. Whenever a new data run or file is loaded, the embedded function is executed, and the new result is displayed.

The syntax for embedding a function in a label is as follows:

text %%*FUNC*(*src*id: *PARAM*):(*FORMAT*)%%

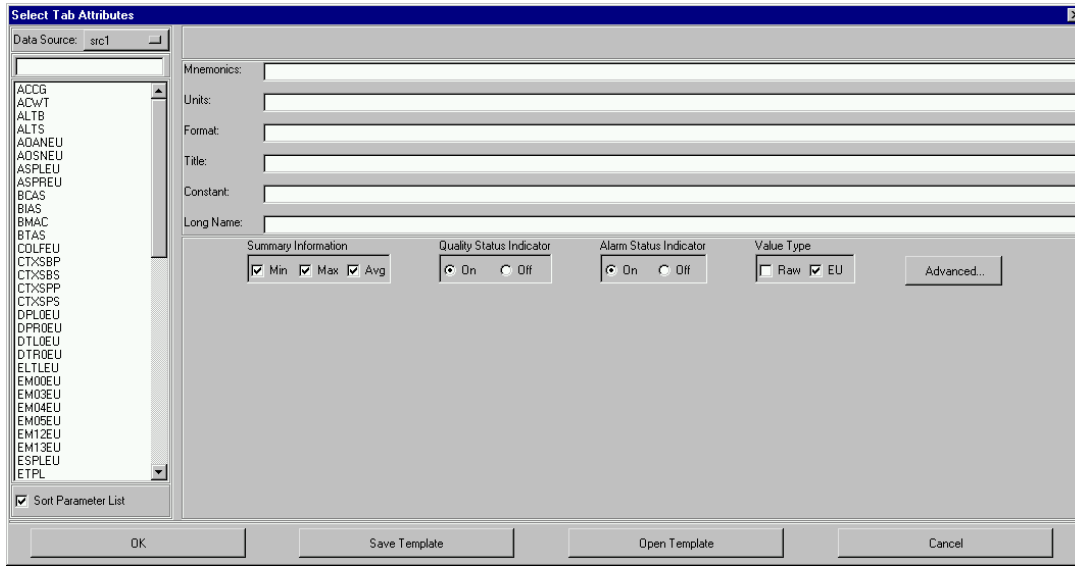
where:

- *text* — Any text you wish to enter.
- *FUNC* — A valid PV-WAVE function (such as AVG)
- *src*id — A vaild TS-WAVE data source id.
- *PARAM* — A valid TS-WAVE parameter.
- *FORMAT* — A FORTRAN format code (such as F8.1)

Example: `The ALTB average is: %%AVG(`*src*`id:ALTB):(F8.3)%%`

Select Tab Attributes Dialog Box

Use this dialog box to select which parameters to write into a tabular data file. In addition, this dialog lets you specify data titles, names of constants, summary information, and data formats.



The Select Tab Attributes dialog box

Parm — Use this text field to locate a specific parameter in the parameter list. Simply type the name of the parameter you wish to locate and press <Return>. If the parameter name you entered is a valid name it is highlighted in the parameter list. Names are case sensitive. If the parameter is not in the list, a message box appears.

Parameter List — Double click on one or more parameter names to select them for output to the tabular data file. When you double click a parameter, its name appears in the **Parameters** field, and its format code appears in the **Format** field.

Parameters — The list of parameters that are to be written to the tabular data file. To add a parameter to this field, double click its name in the scrolling parameter list, or type it directly. If you type the parameter names, be sure to separate them with a space. The names are case sensitive.

Units — Use this text field to enter the units for each parameter, separated by spaces. Each separate unit notation appears above its corresponding column of data in the tabular file.

Format — Use this text field to modify the format code for each parameter listed in the tabular file. By default, the code F10.1 is used. Separate each format code with a space. The default may be modified in TS-WAVE, for more details refer to the README file under the TS-WAVE resource directory.

NOTE You must use standard FORTRAN format codes in this field. For detailed information on FORTRAN format codes, see the *PV-WAVE Programmer's Guide*.

Title 1–6 — Use these text fields to add titles to the file header.

Constant 1–7 — Use these text fields to add constant values to the file header.

Long Name — Use this text field to enter names for the parameters appearing in the tabular data file. Names must be separated by a space; therefore, you must use underscores to string multiple words together.

For example: `Boom_mach_altitude`

NOTE With a format code of 10.1, a long name is truncated after 10 characters. If you want a long name to have more than 10 characters, you must modify the code in the **Format** field.

Summary Information — Lets you select the type of summary information you wish to print for each column of tabular data. For each column, you can print any combination of the minimum, maximum, and average values.

Event Check — Turns the event checking mechanism on or off.

Event checking lets you write data to the tabular file only when a specified event is active. For example, if the parameter specified in the **Event Check Parameter** field records the state "wheels down", then data is only written to the tabular file when this event state is true, or active.

Event Check Parameter — Use this field to enter the name of the event parameter that you wish to check. The name you enter is case sensitive. If **Event Check** is turned on, data is written to the tabular file whenever this event parameter's state is true, or active.

Advanced — Brings up the Select Tab Attributes Advanced Settings dialog box. See *Select Tab Attributes Advanced Settings Dialog Box* on page 91.

OK — Writes the selected parameters to the tabular data file and exits the dialog box.

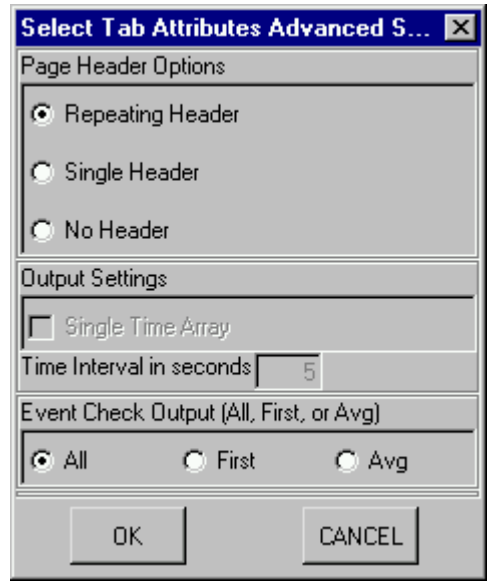
SAVE TEMPLATE — Brings up a standard file selection tool that lets you save a tabular file template. This template saves you the effort required to fill out the Select Tab Attributes dialog box in the future. For example, if you create a tabular data file that consists of 50 out of 200 parameters from a particular run, you can easily save the same parameters from another file if you create a tabular file template. To use a template, you must open it (click **OPEN TEMPLATE**). When the template opens, the Select Tab Attributes dialog is automatically filled in with the appropriate parameters, titles, and other settings.

OPEN TEMPLATE — Brings up a file opening tool that lets you open a previously saved tabular data template file.

CANCEL — Exits the dialog box without taking any action.

Select Tab Attributes Advanced Settings Dialog Box

Use this dialog to specify additional tab file options.



The Select Tab Attributes Advanced Settings dialog box

Page Header Options — Select one of the three options from this checkbox: **Repeating Header**, **Single Header**, or **No Header**. The default is **Repeating**

Header, in which a header appears at the top of every page of a tabular data file. The **Single Header** option specifies that the header only appears at the top of the first page. The **No Header** option specifies that no header information is written to the tabular data file. Only the times and columns of data are written to the file.

CAUTION If you select the No Header option, TS-WAVE can not read the file as a tabular data file. A tabular data file without header information is considered a simple ASCII file. You can, however, read the data file back into TS-WAVE by using the ASCII Previewer. Also, tabular files without a header are easily imported into software packages as ASCII or CSV files.

Output Settings — The default output setting is to write parameters to individual time arrays. A Single Time Array uses only one time array in the tab output.

Time Interval in Seconds — If a Single Time Array is selected, Time Intervals can be set to specify the Interval to create the single Time Array.

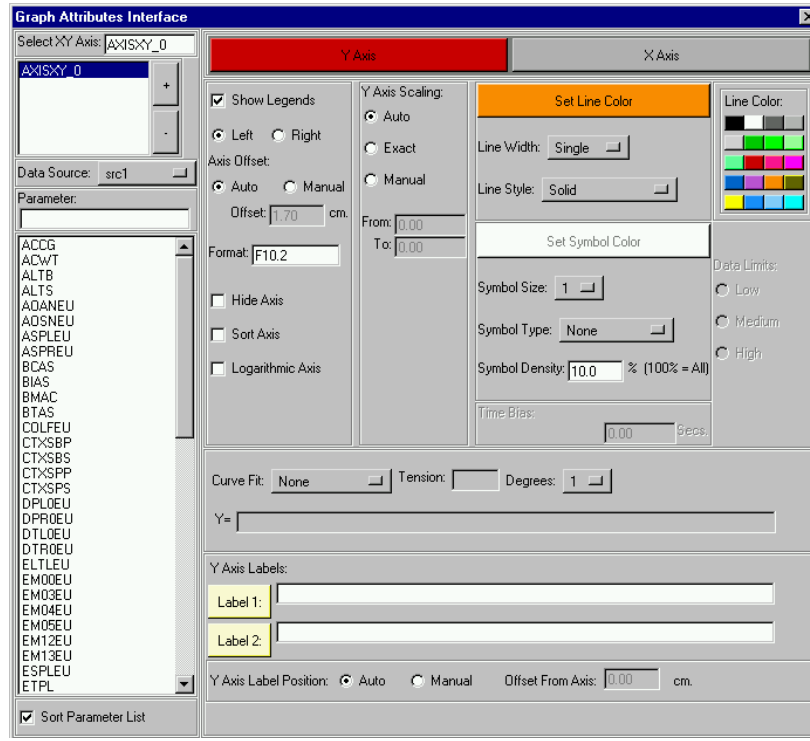
Event Check Output (All, First, or Avg.) — **All** (the default) gives all points of parameter selected when event parameter is active. **First** gives you the first point of the parameter selected when event becomes active. **Avg.** gives you the average of points for each parameter when the event was active.

OK — Apply the settings and exit the dialog box (and return to the Select Tab Attributes dialog box).

CANCEL — Exits the dialog box without taking any action (and returns to the Select Tab Attributes dialog box).

Graph Attributes Interface Dialog Box

Use this dialog to specify the parameters to plot, axes ranges, line characteristics, labels, and other plot attributes.



The Graph Attributes Interface dialog box

Select XY Axis — Allows user to select, add, and delete XY axis. The textbox label displays the currently selected XY axis. The '+' symbol adds an axis, while the '-' symbol deletes an axis.

NOTE Changes are applied to the currently selected XY axis.

Data Source — Use this drop-down menu to select from a list of Data Source IDs. The Paramter list is updated to reflect the available parameters for the source id in which you selected.

Parameter — Use the text field to locate a specific parameter in the parameter list. Simply type the name of the parameter you wish to locate and press <Return>. If the parameter name you entered is a valid name (if it is in the parameter list), it is highlighted in the parameter list. Names are case sensitive. If the parameter is not in the list, a message box appears.

Parameter Info — Double click on a parameter in the Parameterlist to bring up the Parameter Info window. This window displays general and simple statistical information about the currently selected parameter. If a parameter is not selected, clicking on the Parameter Info button will have no effect.

Parameter List — Click on a parameter name to select it for plotting on the currently selected *y*-axis. When you click a parameter, its name appears in the **Label 1** field.

Sort Parameter List — This check box determines if the parameters within the data file are sorted in an alphabetically ascending order for the **Parameter List**. The default is to sort the parameter names. If you wish to display the list of parameters as they actually appear in the data file, click on this box.

NOTE the Data Type option is displayed only for certain type of data files. This is site and data file specific. Depending on your type of data, this option may not be an active part of the Graph Attributes Interface.

Y-Axis Configuration

The following functions are grouped in the upper section of the dialog box. These functions are mainly used to configure the appearance of *y*-axis and plot lines.

Axis Offset — This text field specifies the offset in centimeters for the current axis from the edge of the plot or the previous axis location. If you are in Auto Axis mode (the default), the Axis Offset is based on the width of the axis Format statement.

Hide Axis — Allows a user to disable the drawing of the current axis. Other defined axis attributes (line color, symbol type, etc.) are applied to the parameter associated with the hidden axis. The axis itself is not displayed.

Sort Axis — This option forces the data associated with the current axis to be sorted in ascending order prior to being displayed. This is the default behavior for TS-WAVE.

Format — Use this text field to modify the format code for the tick mark values on the *y*-axis of the currently selected graph. By default, the code F8.1 is used.

NOTE You must use standard FORTRAN format codes in this field. For detailed information on FORTRAN format codes, see the *PV-WAVE Programmer's Guide*.

Align Axis — Positions the currently selected axis on the left or the right side of the graph.

Y-Axis Scaling — Select **Auto** to use the axis range stored in the parameter; select **Manual** to specify the axis range yourself. If you select **Manual**, the **From** and **To** text fields become editable.

From — Lets you specify the initial value on the *y*-axis. This field is only active when **Manual** is selected in the **Y-axis Scaling** button box.

To — Lets you specify the end value on the *y*-axis. This field is only active when **Manual** is selected in the **Y-axis Scaling** button box.

Curve Fit — Lets you select a curve fitting algorithm to apply to the plotted points.

Choices are:

- **None** — Do not use any curve fitting. (Default)
- **User Equation** — If this method is selected, you can enter an equation in the **y=** field immediately below this menu.
- **Poly Fit** — Produces an *n*-degree polynomial curve through the set of data points using the least-squares method. Select the degree of the polynomial to be fitted to the data with the **Degrees** option menu. For more information on the Poly Fit algorithm, refer to the POLY_FIT function in the *PV-WAVE Reference*.
- **Poly FitN** — Fits an *n*-variate polynomial to some *n*-dimensional data points using the least-squares method. Select the degree of the polynomial to be fitted to the data with the **Degrees** option menu. For more information on the Poly FitN algorithm, refer to the PV-WAVE Users' Library routine POLYFITN.
- **Poly FitW** — This algorithm is similar to the Poly Fit algorithm, except that it permits the weighting of data points. Select the degree of the polynomial to be fitted to the data with the **Degrees** option menu. For more information on the Poly FitW algorithm, refer to the POLYFITW function in the *PV-WAVE Reference*.
- **Spline** — Performs a cubic spline interpolation of the parameter's data points. For more information on the Spline algorithm, refer to the SPLINE function in the *PV-WAVE Reference*.

Tension — Lets you enter the amount of tension to be put on the spline curve. This value controls the smoothness of the fitted curve. This text field is active when the **Spline** function is selected.

Degrees — Lets you pick the degree of the polynomial to be fitted to the data. This menu is active when the **Poly Fit** functions are selected.

y = — Enter any PV-WAVE expression in this field. For example, `y=sin(x)` is a valid user equation. This text field is active when **User Equation** is selected from

the **Curve Fit** menu. Note that when you select an algorithm from the **Curve Fit** menu, its equation is placed in this text field.

Line Color — Lets you pick a line color for the current *y*-axis variable plot.

Line Width — Lets you specify the thickness of plot lines. Widths include: **Single**, **Double**, and **Triple**.

Line Style — Lets you specify the style of plot lines. Styles include: **None**, **Solid**, **Short Dashes**, **Long Dashes**, **Long-short Dashes**, and **Long-short-short Dashes**.

Symbol Color — Lets you pick a color for plot symbols.

Symbol Size — Lets you pick a size for the plot symbols.

Symbol Type — Lets you pick a symbol type for the current *y*-axis variable plot. Types include: **None**, **Plus** (+), **Asterisk** (*), **Period** (.), **Diamond**, **Triangle**, **Square**, **X**, and **User Defined**.

NOTE If you wish to add user-defined symbols to the **Symbol Type** menu, please contact Visual Numerics Customer Support for more information.

Symbol Density — This text field allows a user to specify a percentage that defines how often symbols are drawn on a data line. An entry of 50 implies a symbol is placed at every other data value location. An entry of 33 places a symbol at every third data value location. If the entry is 100, every actual data value location is indicated with a symbol.

Label Text Fields

These text fields appear at the bottom of the dialog box. They are mainly used to add text labels to the *y*-axes and to modify the appearance of the text (fonts, text color, text size, and so on).

Label 1 — Click the **Label 1** button to bring up the Advanced Label Editing Interface dialog box. This dialog lets you customize the appearance of the text entered in the adjacent field. You can set the font (normal, italic, bold, bold italic, Greek, and math), character position normal (normal, superscript, subscript), print size (1 is the default, 2 is twice as big, and so on), orientation (90 degrees is the default). For more discussion on the Advanced Label Editing Interface, see page 102.

NOTE The text formatting commands that are inserted into the label in the Advanced Label Editing Interface dialog correspond to the text formatting com-

mands used in PV-WAVE. For detailed information on PV-WAVE text formatting commands, see the *PV-WAVE User's Guide*.

Label 2 — Click the **Label 2** button to bring up the Advanced Label Editing Interface dialog box, described previously.

The selected parameter name is automatically copied into the **Label 1** text field (however, you can change this name if you wish). In other words, the parameter name is normally the first label to appear on the *y*-axis.

X-Axis Configuration

The following functions are available when you click on the X Axis tab at the far top right of the Graph Attributes Interface dialog box. If the behavior or options differ from the Y Axis options, they are explained in this section.

Select XY Axis — Allows user to select, add, and delete XY axis. The textbox label displays the currently selected XY axis. The '+' symbol adds an axis, while the '-' symbol deletes an axis.

NOTE Changes are applied to the currently selected XY axis.

Data Source — Use this drop-down menu to select from a list of Data Source IDs.. The Paramter list is updated to reflect the available parameters for the source id in which you selected.

Time Series Plot — This check box determines if TS-WAVE parameters are displayed in a Time Series mode or an X-Y mode. The default is Time Series Mode. Click on this box if you want to operate in X-Y mode. While in X-Y mode, the **Parameter List** becomes active and enables the specification of any parameter list to be used as the X Axis values.

X-Axis Scaling — Select **Auto** to scale the *x*-axis to the range of values stored in the parameter; select **Manual** to specify the axis range yourself. If you select **Manual**, the **From** and **To** text fields become editable.

From — Lets you specify the initial value on the *x*-axis. This field is only active if **Manual** is selected in the **X-axis Scaling** button box.

To — Lets you specify the end value on the *x*-axis. This field is only active if **Manual** is selected in the **X-axis Scaling** button box.

Format — Use this text field to modify the format code for the tick mark values on the *x*-axis of the currently selected graph. By default, the code F5.1 is used.

NOTE You must use standard FORTRAN format codes in this field. For detailed information on FORTRAN format codes, see the *PV-WAVE Programmer's Guide*.

Start Time and Stop Time — Depending on the specific data format used with TS-WAVE, either the **From**, **To** and **Format** fields or the **Start Time**, **Stop Time** fields are active. Similar to the **From** and **To** definitions, the **Start** and **Stop** fields allow the control of the X-Axis range while in Manual mode for X-Y plotting.

Units — Lets you choose minutes or seconds as the unit spacing along the *x*-axis. This is a site-specific option based on the data to be displayed.

X-Axis Labels — TS-WAVE allows the display of two text labels on the X-axis. The labels are referred to as Label 1 and Label 2.
Label 1 — If the Time Series Plot check box is selected, the default for Label 1 is Time. If in X-Y mode, the default is the current selected parameter. You may change the text in this field if you like.
Label 2 — There is no default for the Label 2 text field. Enter any text to be displayed as the second X-axis label.

Show Legends — If you wish to display line legends, select **On**. Line legends appear above the graph object. Line legends identify the type of line (solid, dashed, and so on) and type of plotting symbol associated with a particular variable.

Hide X-Axis — If you do not wish to display the *x*-axis on the graph, select **Off**.

Sort X Data — This check box specifies if the selected X data is to be sorted prior to display of the data on the selected graph object. The default is to sort the X data in ascending numerical order.

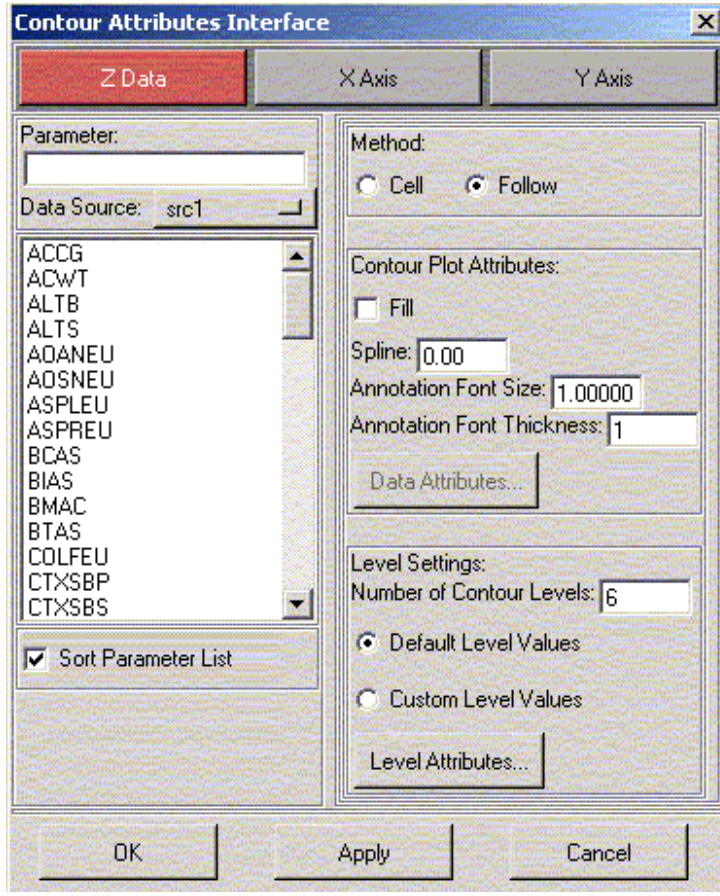
OK — Plots the specified parameter and exits the dialog box.

Apply — Plots the specified parameter, but does not exit the dialog box.

Cancel — Exits the dialog box without taking any action.

Contour Attributes Interface Dialog Box

Use this dialog to specify the parameters to plot, axes ranges, method options, line characteristics, labels, and many more plot attributes.



The Contour Attributes Interface dialog box

Contour creates a contour plot using TS-WAVE data stored in three one-dimensional arrays. Each of the X, Y, and Z tabs must be assigned parameters before selecting OK or Apply. Each of the data sets must be of identical size and contain three or more data points.

Z Data, X Axis, and Y Axis Tabs

Z Data Tab — The Z Data parameter is the array of values that will make up a contour surface. Select a parameter or type in the name of an existing parameter to assign Z data.

X Axis Tab — The X Axis parameter is the array of values that will be used to plot the X coordinates for the contour surface. Select a parameter or type in the name of an existing parameter to assign the X data.

Y Axis Tab — The Y Axis parameter is the array of values that will be used to plot the Y coordinates for the contour surface.Select a parameter or type in the name of an existing parameter to assign the Y data.

Z Data Attributes Dialog

Method Options

Contour can draw contours using one of two different methods: Cell or Follow. Although these two methods both draw correct contour maps, differences in their algorithms can cause small differences in the resulting plot.

Cell Method — The cell-drawing method examines each array cell and draws all contours emanating from that cell before proceeding to the next cell. This method is efficient in terms of computer resources but does not allow contour labeling.

Follow Method — The line-following method searches for each contour line and then follows the line until it reaches a boundary or closes. This method gives better looking results, especially with non-solid linestyles, and allows contour labeling, but requires more computer time.

Contour Plot Attributes Options

Fill — Fill the contours with colors defined in the Level Attributes dialog.

NOTE The Fill attribute is available only for the Follow method.

Spline — Specifies that contour paths are to be interpolated using cubic splines. The appearance of contour plots of arrays with low resolution may be improved by using spline interpolation. In rare cases, contour lines that are close together may cross because of interpolation. Splines are especially useful with small data sets (less than 15 array dimensions). With larger data sets the smoothing is not as noticeable, and the expense of splines increases rapidly with the number of data points.

The Spline value specifies the length of each interpolated line segment in normalized coordinates. The default value is 0.005. Smaller values for this parameter yield smoother lines, up to the resolution of the output device, at the expense of more computations.

NOTE The Spline attribute is available only for the Follow method.

Annotation Font Size — Sets the overall character size for contour annotation.

NOTE This attribute is available only for the Follow method.

Annotation Font Thickness — Sets the thickness of characters drawn.

NOTE This attribute is available only for the Follow method.

Data Attributes Button — Brings up the Data Attributes dialog box.

Data Attributes Dialog — It is recommended that, before adjusting any of the data attributes, you generate an initial plot using the defaults to give you a good starting point.

The attributes in this dialog will be applied to the original data to aid in the reduction of computer time required to generate contour plots.

Weight Order — A scalar specifying the order of the weighting function. The dependent variable at a point is computed as a weighted average of the variable over all neighborhood data points. The weighting function is 1/e^w where e is the Euclidean distance between the grid point and the data point and w is the Weight Order of the function. This value should be 0<=weight<=5. The default value is 2.

Neighborhood Size — A scalar between 0 and 1 specifying neighborhood size. A value of one gives a maximal neighborhood which includes all data points, while lower values yield smaller neighborhoods. This value should be 0<=neighborhood<=1. The default value is 1.

X and Y Grid Sizes — Final resampled dimensions of regridded data. The higher the grid dimensions, the more computing time is required. The values entered here should be 0<=gridsize<=1000.

Resample Data Factor — The value of this slider ranges from 1 to zdatasize/3. By default, if at least one plot has not been generated, the factor will range from 1 to 500. Use this value to reduce the total size of your dataset for quicker plotting. The value entered here will be the factor by which the total number of datapoints will be resampled before plotting. For example, if you enter 2, your dataset will be resampled to half its original size.

Level Settings

Number of Contour Levels — The number of equally-spaced contour levels that are produced. The maximum allowed is 150.

Default Level Values — Calculate default contour level values. These values can be viewed in the Level Attributes dialog by first generating a plot using Default

Level Values, then selecting Custom Level Values and clicking the Level Attributes button.

Custom Level Values — Assigns custom level values for each contour. Values can be assigned by clicking on the Level Attributes button.

Level Attributes Button — Brings up the Level Attributes dialog box.

Level Attributes Dialog

Contour # — The attributes at the left will be applied to the contour selected in the Contour # column. The number of levels that appear in this list are defined by the Number of Contour Levels field in the Contour Attributes Interface dialog.

Default Annotation — Label the selected contour with its Z value.

No Annotation — Do not label the selected contour.

Use Special Annotation — Label the selected contour using the string defined in the Special Annotation field.

Special Annotation — The string entered in this field will be used to label the selected contour when Use Special Annotation is selected.

Custom Level Value — Specifies the contour level assigned to the selected contour. This field is available when Custom Level Values is selected in the Contour Attributes Interface dialog box.

Line Style — Specifies the line style to use for the selected contour.

Line/Fill Color — Defines the color used to draw the selected contour, or the fill color for filled contours. To select a color, either click on a color, or type in the color index, from 0-19, and press Enter.

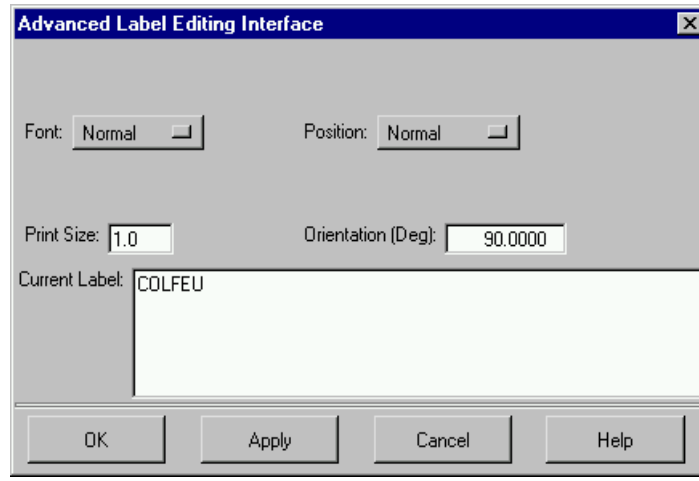
Line Thickness — Specifies the thickness of lines used to draw the selected contour level.

Advanced Label Editing Interface Dialog Box

Use this dialog box to modify the appearance of the text in a label.

TIP The functions in this dialog insert PV-WAVE text formatting commands into the label string (for example, !8 for italic text). These text formatting commands are explained in the *PV-WAVE User's Guide*.

NOTE A formatting command (such as !8 for Italic) must precede the text it is supposed to change. Therefore, we recommend that you first select a font or position, and then enter the text.



The Advanced Label Editing Interface dialog box

Font — Lets you select the font.

Position — Lets you select the position of the text.

Current Label — This text field reflects the current label string. You can enter text and formatting commands directly in this field if you wish.

Weight — Lets you select the weight of the text.

Print Size — Lets you select the size of the text when printed. This is a relative size, where a value of 2 is twice as big as 1.

Orientation — Lets you enter the number of degrees to rotate the text, where 0 is horizontal.

Justify — Lets you select **Left**, **Center**, or **Right** justification.

OK — Applies the text changes, and exits the dialog box.

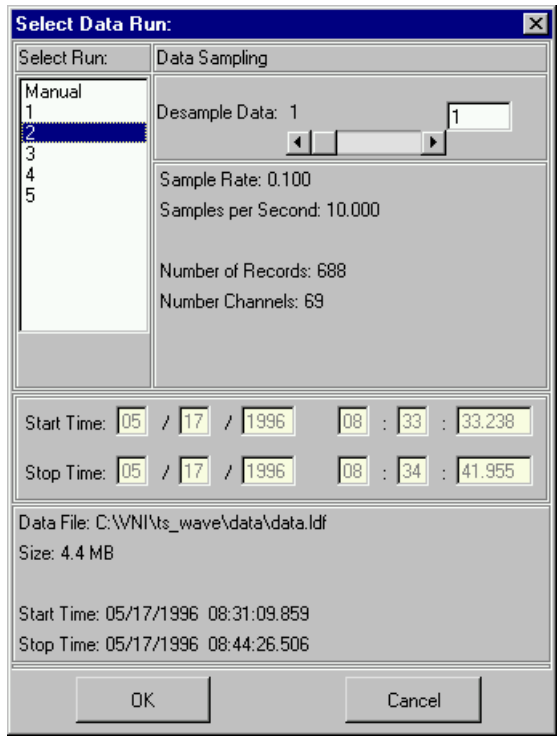
Apply — Applies the text changes, but does not exit the dialog box.

Cancel — Exits the dialog box without taking any action.

Select Data Run Dialog Box

Use this dialog to specify which run number to plot and the desampling rate.

NOTE LDF only - your site may be different based on your actual data file format and contents of the data file.



The Select Data Run dialog box

Select Run — Click on the run number that you wish to examine from the data file. Click **Manual** to examine all of the data. (If you are using this dialog to create a batch file or to create a multi-run tab file, you can select multiple runs.)

Start Time — Displays the start time-stamp for the selected run. You can edit this field when **Manual** is selected. This provides a way to subset your data. For instance, you can examine a 3.25 minute segment of the entire flight by specifying exact start and stop times.

Stop Time — Displays the stop time-stamp for the selected run. As with **Run Start**, you can edit this field when **Manual** is selected.

DeSample Data — Lets you specify the interval at which to read parameters from the data file. A desampling rate of 1 reads all parameters, a desampling rate of 2 reads every second parameter, and so on.

Sample Rate — (non-editable) Reports the frequency at which the data was collected.

Samples per Seconds — (non-editable) Reports the number of seconds.

Number of Records — (non-editable) Reports how many records are to be displayed given the current run or how many records are to be displayed given the specified start and stop times if in Manual mode.

Number of Channels — (non-editable) Reports the number of parameters that were collected in the run, or the number of parameters if the file is in Manual mode.

Data File — (non-editable) Displays the full pathname of the currently selected data file.

Size — (non-editable) Displays the size of the currently selected data file.

Start Time — (non-editable) Displays the start time of the entire file.

Stop Time— (non-editable) Displays the stop time of the entire file.

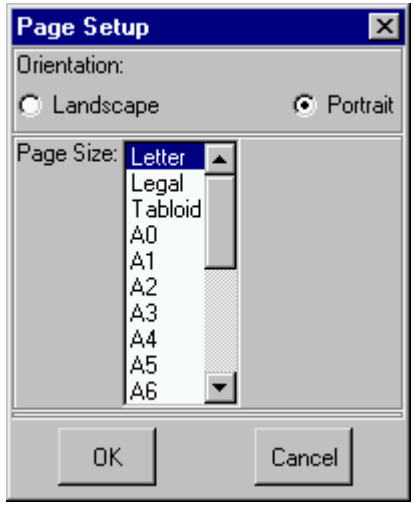
OK — Loads the selected run and exit the dialog box.

Apply — Loads the selected run, but do not exit the dialog box.

Cancel — Exits the dialog box without taking any action.

Page Setup Dialog Box

Use this dialog to change the orientation and size of the drawing area.



The Page Setup dialog box

Portrait — If this option is selected, the drawing area is displayed in portrait orientation, with the *x*-axis along the short dimension of the page.

Landscape — If this option is selected, the drawing area is displayed in landscape orientation, with the *x*-axis along the long dimension of the page.

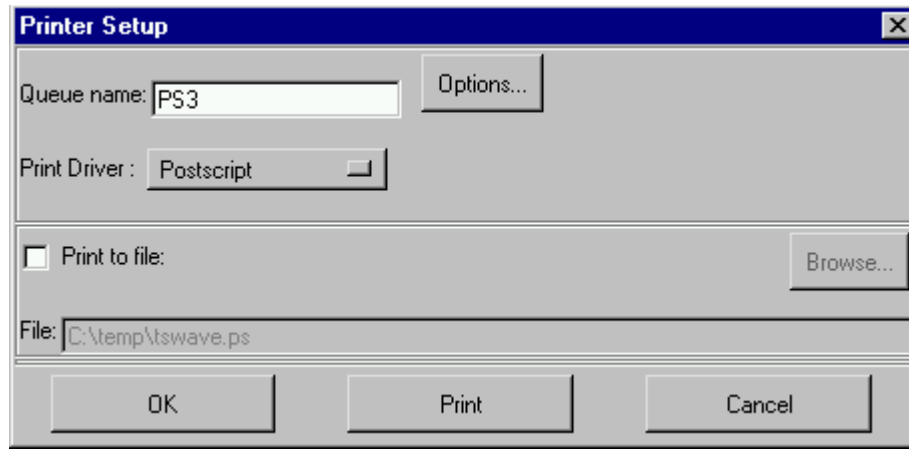
Page Size — Selects the output paper size from the list. Letter is 8.5 by 11 inches. Legal is 8.5 by 14 inches. Tabloid is 11 by 17 inches. The A and B sizes adhere to their respective international paper sizes.

OK — Applies the changes and exit the dialog box.

Cancel — Exits the dialog box without applying any changes.

Printer Setup Dialog Box

This dialog lets you specify a printer or type of file to print to.



The Printer Setup dialog box

Options — Brings up a dialog box that lets you specify options for the currently selected **Printer Type**. For detailed information about the options for each device, refer to *Appendix B, Output Devices and Window Systems*, in the *PV-WAVE Reference*.

Queue Name — Enter the name of the printer in this text field. The default printer name is specified in the resource file `tswave_print.ads`. For more information on this resource file, see *Resource Files* on page 111.

Printer Driver — Selects the type of printer or print file that you wish to use. Choose the **Options** function to specify additional settings for a device or file. Choices are:

- PostScript — Sends output to a PostScript file.
- Windows MetaFile — Saves graphics in a metafile. Select **Print to file** and specify a name for the metafile. A metafile is a standard Enhanced Metafile format that allows files to be used in other applications that support metafiles, such as Microsoft Word and Microsoft Excel.
- CG Metafile— Sends output to a Computer Graphics Metafile device. If **Print to file** is selected, the plot is saved in a CGM file.
- HP PCL — Sends output to a Printer Control Language device. If **Print to file** is selected, the plot is saved in a PCL file.
- HP GL — Sends graphics to an HPGL device. If **Print to file** is selected, the plot is saved in an HPGL file.

Print to file — Select this checkbox if you wish send the print job to a file instead of a printer. The saved file can be printed on an appropriate device, but cannot be read back into an TS-WAVE session. You can enter a simple filename (in which case the file is saved in the current working directory) or a complete pathname in the text field.

Browse — Brings up a standard file saving tool that lets you enter a filename and choose a directory in which to save the print file.

OK — Sets the printer and exits the dialog box.

Print — Sends the graphics to the specified print device or file.

Cancel — Exits the dialog box without taking any action.

Data Zoom Dialog Box

Data Zoom feature lets you see plotted data up close.

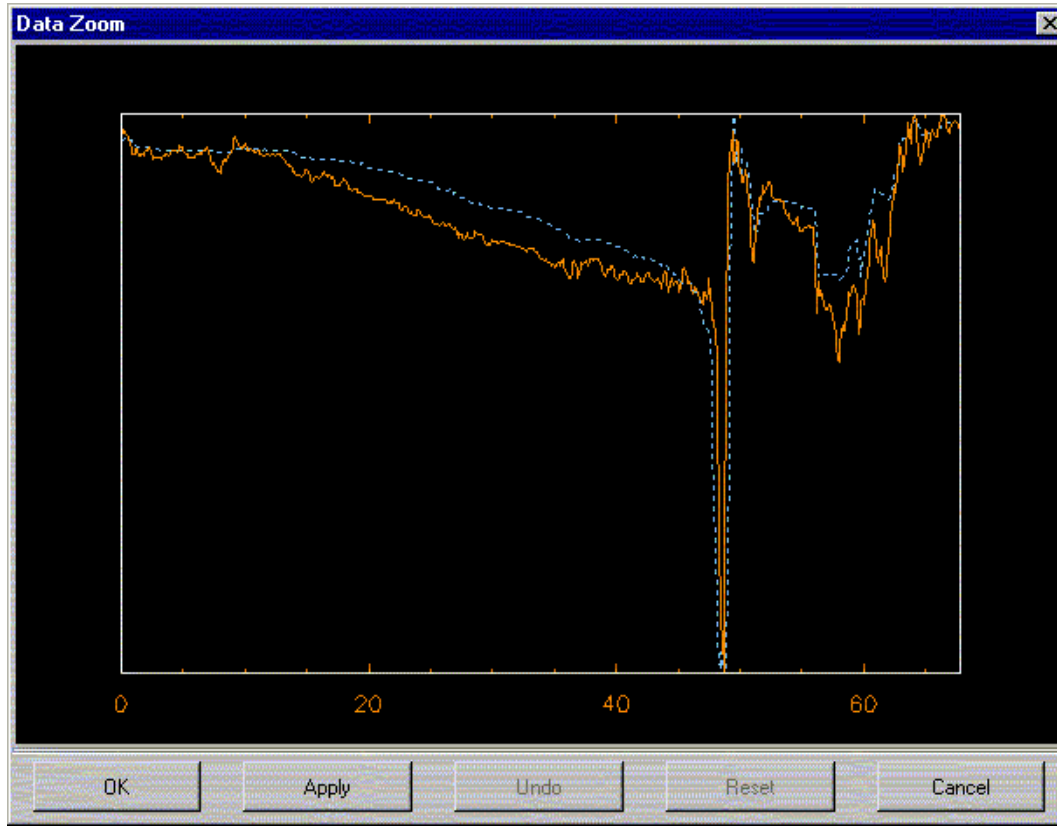
Opening the Data Zoom Window

To open the data zoom dialog, select one or more graph objects displaying at least one parameter. Then select the **View=>Data Zoom** menu item. To see the Data Zoom dialog box, turn to page 21.

The following criteria must be met in order to use the data zoom dialog:

- There must be at least one graph object selected.
- The selected graph object(s) must be displaying at least one parameter.
- If more than one graph object is selected, all graphs must have the same X range. You can use the **Edit=>Align Graph** menu item to ensure all graph objects have the same X range.
- All graph objects must be Time-History plots. Data Zoom is not available for X-Y plots.

NOTE Parameters may be displayed using any Y range. Each will be displayed and scaled within its own range.



The Data Zoom dialog box

When the Data Zoom window opens, the parameter(s) appear plotted on a single set of axes. If more than one graph object was selected, there will be no labels on the Y axis since the axis represents more than one scale. If the parameter was given a manual range or it had been previously zoomed, this range is used in the data zoom window.

Parameters are displayed with the same line color and characteristics as in the original graph object.

TIP To reset the original range for all selected graph objects, use the **View=>Set Auto Scale** menu item. If you want to reset the original range on a single zoomed graph object, double click on the graph object to bring up the Graph Attributes dialog and set Y Axis Scaling to “Auto.”

Mouse Operations

Clicking and dragging the left mouse button in the Data Zoom Window causes a rubber band box to be drawn. When the mouse button is released, the data is zoomed to the range defined by the rubber band box. The data can be further zoomed by repeating the above procedure.

Keyboard Operations

Once the data has been zoomed, the following keys are active:

Left/Right Arrow — Pans the plot to the left or right.

Up/Down Arrow — Pans the plot up or down.

Home — Moves directly to the beginning of the original range.

End — Moves directly to the end of the original range.

The following modifier keys affect the amount by which the Up/Down/Left/Right keys pan within the range:

Shift — Allows the plot to be panned at an increased rate. Use this to pan to a new region of interest in plot.

Ctrl — Allows the plot to be panned at a decreased rate. Use this to fine tune the displayed range.

Dialog Buttons

OK — Applies the zoomed range(s) to the selected graph object(s) and dismisses the dialog.

Apply — Applies the zoomed range(s) to the selected graph object(s) without dismissing the dialog.

Undo — Reverts to the previous zoomed range(s) or the original range(s) if it was the first zoom.

Reset — Reverts to the original zoomed range(s), regardless of how many zooms/ applies have been performed.

Cancel — Exits the dialog without saving the current scales. If an Apply has already been done, it will NOT undo these changes.

Miscellaneous Topics

This section discusses the ASCII PREVIEWER and resource files.

ASCII PREVIEWER

The ASCII PREVIEWER is a tool for interactively providing information about the organization of an external ASCII data file. Use this tool to view a file's contents and select which parts of the file are to be read in as TS-WAVE parameters.

TIP The ASCII PREVIEWER is designed to work with ASCII files that have a record separator, such as a carriage return/line feed or newline, at the end of each line. Furthermore, the ASCII PREVIEWER can only be used to read column-oriented files containing 1D or 2D data.

The ASCII PREVIEWER works like the WzPreview Tool, one of the VDA Tools found in PV-WAVE. For detailed information on this tool, select **Help=>On Window** from the ASCII PREVIEWER menu bar.

For more information on how to use the WzPreview Tool, see the *PV*-WAVE Tutorial.

Resource Files

Many default settings for TS-WAVE are specified through resource files.

These resource files are located in:

```
<install_dir>\TSWAVE\resource
```

where `<install_dir>` is the drive and directory where TS-WAVE is installed. For example:

```
C:\applications\TSWAVE\resource
```

You can modify these resource files using any text editor. You must exit and restart TS-WAVE for your changes to take effect.

For example, to change the default print, paper size and orientation for the printed output, open the resource `tswave_print.ads` and change the lines:

```
Default_Printer:  HP4MP2

Default_PaperSize:Letter
```

```
Default_Orientation:Portrait
```

There are numerous resources associated with TS-WAVE. Each file addresses a particular section of TS-WAVE functionality. Resource files allow you to customize the look, language and execution of the overall application.

TS-WAVE Index

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W

X

Z